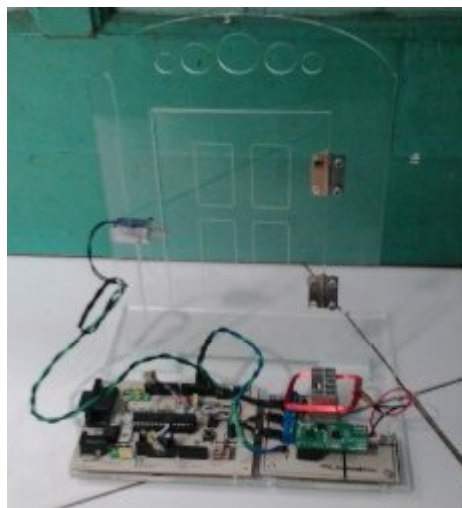




**ALAT PENGAMAN PINTU RUMAH MENGGUNAKAN RFID
(*RADIO FREQUENCY IDENTIFICATION*) 125 KHz
BERBASIS MIKROKONTROLER ATMEGA328**

PROYEK AKHIR

Diajukan kepada Fakultas Teknik Universitas Negeri Yogyakarta untuk
Memenuhi Sebagian Persyaratan Guna Memperoleh
Gelar Ahli Madya Teknik



Oleh :

DIDIK SUYOKO

NIM. 09507131032

**PROGRAM STUDI TEKNIK ELEKTRONIKA
FAKULTAS TEKNIK
UNIVERSITAS NEGERI YOGYAKARTA
2012**

PERSETUJUAN

Proyek akhir yang berjudul “**ALAT PENGAMAN PINTU RUMAH MENGGUNAKAN RFID (*Radio Frequency Identification*) 125 KHz BERBASIS MIKROKONTROLER ATMEGA328**” ini telah disetujui oleh pembimbing untuk diujikan.

Yogyakarta, 28 November 2012

Menyetujui,

Dosen Pembimbing Proyek Akhir

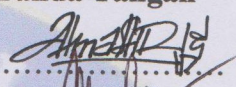
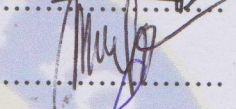
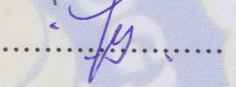


Yuniar Indrihapsari, M. Eng
NIP.19820621 201012 2 002

PENGESAHAN


Proyek akhir yang berjudul “ ALAT PENGAMAN PINTU RUMAH MENGGUNAKAN RFID (*RADIO FREQUENCY IDENTIFICATION*) 125 KHz BERBASIS MIKROKONTROLER ATMEGA328 ” ini telah dipertahankan didepan dewan penguji pada tanggal 7 Desember 2012 dan dinyatakan lulus.

Dewan Penguji

Nama	Jabatan	Tanda Tangan	Tanggal
Yuniar Indrihapsari, ST., M.Eng.	Ketua Penguji		7/12/2012
Drs. Muhammad Munir, M. Pd.	Sekretaris Penguji		13/12/2012
Drs. Masduki Zakaria, M.T.	Penguji		13/12/2012

Yogyakarta, 17 Desember 2012
Fakultas teknik
Universitas Negeri Yogyakarta



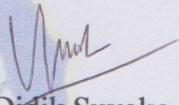

Dr. Moch. Bruri Triono
NIP.19560216 198603 1 003

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa dalam Proyek Akhir ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar Ahli Madya atau gelar lainnya di suatu perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis oleh orang lain, kecuali secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.



Yogyakarta
Yang Menyatakan,


Didik Suyoko
NIM. 09507131032

PERSEMBAHAN

Dengan segala kerendahan hati penulis persembahkan karya sederhana ini kepada:

Bapak dan Ibu tercinta yang telah memberikan kasih sayang, perhatian,
dorongan dan doanya yang tiada henti.

Yang tercinta Istriku Hesti Nugraheni dan saudaraku (Mbak Yayuk, Mas Didik,
Mbak Ebit, Mbak Susi, Drajat Nugroho, Lia, dan Jeki) mertuaku tersayang serta
ponakanku tersayang.

(John Paul Ipunk, Vian Prayoga, Kerbau, Du-Blink) terima kasih untuk semua
semangat dan senyum, serta canda tawanya, kalian sahabat
terbaik yang pernah aku kenal.

My friend Kelas B (Irwan, Rian, Taufik, Bobby, Bebek, Desi, Hunny dan
semuanya) terima kasih atas kebersamaannya, kepercayaan dan kenangan suka
serta duka kalian tidak akan pernah kulupa.

Buat sobat cokeiwa (Tembaz, Ocym, J-Gog, Tlotox) thanks for your spirit,

Buat teman-temanku kelas D2, terima kasih atas do'a restu kalian.

Almamater UNY tercinta.

MOTTO

“Tugas kita bukanlah untuk berhasil. Tugas kita adalah untuk mencoba, karena didalam mencoba itulah kita menemukan dan belajar membangun kesempatan untuk berhasil”.

“Dan bersabarlah dalam menunggu ketetapan Tuhanmu, maka sesungguhnya kamu berada dalam penglihatan Kami, dan bertasbihlah dengan memuji Tuhanmu ketika kamu bangun berdiri”.

(Q.S. Ath Thuur 48)

“Awalilah harimu dengan senyum
Semangat baru untuk hari yang baru
Yakinilah bahagia kan datang
Jangan pernah menyerah tuk hadapi kerasnya hidup ini
Mantapkanlah hatimu tuk melangkah kedepan lebih pasti
Buanglah rasa ragu yang menghalangi dan selalu membelenggu
Lakukanlah niatmu dan katakan AKU HARUS BISA”

(*Optimis*_John Paul Ipunk)

**PENGAMAN PINTU RUMAH MENGGUNAKAN RFID
(*RADIOFREQUENCY IDENTIFICATION*) 125 KHz
BERBASIS MIKROKONTROLER ATMEGA 328**

Oleh: Didik Suyoko
NIM : 09507131032

Abstrak

Tujuan pembuatan pengaman pintu rumah menggunakan sensor RFID (*Radio Frequency Identification*) 125 KHz berbasis mikrokontroler ATmega 328 adalah bertujuan membuat suatu alat pengaman pintu yang mudah, murah, praktis dalam penggunaan untuk dapat meningkatkan kenyamanan dan keamanan dalam membuka pintu rumah tanpa harus memegang bermacam-macam kunci yang mungkin sangat mengganggu. Alat ini akan mendeteksi/bekerja setelah sensor RFID mendeteksi *tag card* yang dihadapkan, yang secara otomatis akan membaca *tag card* untuk membuka pengunci pada pintu yang berupa solenoid.

Metode yang digunakan dalam pembuatan pengaman pintu rumah menggunakan sensor RFID (*Radio Frequency Identification*) 125 KHz berbasis mikrokontroler ATmega 328 ini adalah eksperimental. Metode ini terdiri dari beberapa tahap yaitu (1) Identifikasi kebutuhan, (2) Analisis Kebutuhan, (3) Perancangan perangkat keras dan perangkat lunak, (4) Pembuatan alat, (5) Pengujian Alat dan (6) Pengoperasian Alat. Perangkat keras terdiri dari (1) Sistem minimum ATmega328 sebagai pengendali utama, (2) *tag card* sebagai kunci untuk membuka pengaman pintu, (3) solenoid sebagai pengunci pintu.

Berdasarkan hasil pengujian dapat disimpulkan bahwa pengaman pintu ini dapat bekerja sesuai dengan prinsip kerja yang dirancang. Hal tersebut ditunjukkan oleh semua rangkaian pada saat bekerja, sensor RFID dapat mendeteksi *tag card* dengan *databasenya*, solenoid juga mampu membuka penguncinya ketika *database tag card* sudah dicocokkan oleh mikrokontroler. Prosentase error pada pengukuran tegangan solenoid ini sebesar 15,8%. Prosentase error pengukuran tegangan pin 13 sebesar 9,8%. Prosentase error pengukuran tegangan pada modul RFID sebesar 2. Dan prosentase error pada regulator sebesar 0%.

Kata Kunci : Sensor RFID, ATmega328, *Tag Card*, Pengaman Rumah

KATA PENGANTAR



Puji syukur penulis panjatkan kehadirat Allah SWT, yang telah melimpahkan rahmat, taufik dan karunia-Nya, sehingga dapat menyelesaikan Laporan Proyek Akhir yang berjudul “Alat pengaman pintu rumah menggunakan RFID (*radio frequency identification*) 125 KHz berbasis mikrokontroler ATmega 328”. Tujuan dari penyusunan Proyek Akhir ini adalah sebagai syarat kelulusan pada program studi Teknik Elektronika D3 Universitas Negeri Yogyakarta.

Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan laporan Tugas Akhir ini tidak akan lancar. Oleh karena itu pada kesempatan ini, izinkanlah penulis menyampaikan ucapan terima kasih kepada:

1. Bapak Dr. Moch. Bruri Triyono, selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta.
2. Bapak Drs. Muhammad Munir, M.Pd selaku Ketua Jurusan Pendidikan Teknik Elektronika.
3. Bapak Drs. Djoko Santoso, M.Pd. selaku Ketua Program Studi Teknik Elektronika dan Koordinator Proyek Akhir.
4. IbuYuniar Indrihapsari, ST., M.Eng. selaku Pembimbing.
5. Seluruh Dosen dan Karyawan di Jurusan Teknik Elektronika Fakultas Teknik Universitas Negeri Yogyakarta. Yang telah mendidik dan memotivasi selama kuliah di UNY.
6. Keluarga besar yang telah memberikan kasih sayang dan motivasi selama ini.
7. Teman-teman mahasiswa Teknik Elektronika UNY angkatan 2009.
8. Semua pihak yang tidak dapat disebutkan satu per satu yang telah membantu dalam penyelesaian laporan ini.

Akhirnya disadari sepenuhnya bahwa dalam penyusunan Proyek Akhir ini masih jauh dari kesempurnaan, sehingga saran, masukan, dan kritik sangatdiperlukan demi

kesempurnaan, dan semoga penyusunan Proyek Akhir ini dapat memberikan kontribusi bagi semua pihak.

Yogyakarta, November 2012

Didik Suyoko

DAFTAR ISI

	halaman
HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN KEASLIAN	iv
HALAMAN PERSEMBAHAN	v
MOTTO	vi
ABSTRAK	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xiv
DAFTAR LAMPIRAN	xvi
BAB I PENDAHULUAN	1
A. Latar Belakang Masalah	1
B. Identifikasi Masalah	4
C. Batasan Masalah	4
D. Rumusan Masalah	5
E. Tujuan	5
F. Manfaat	6
G. Keaslian Gagasan	7
BAB II PENDEKATAN MASALAH	8
A. RFID (<i>Radio Frekuensi Identification</i>)	8
1. Pembaca RFID	9
2. <i>Tag</i> RFID	9

3. Frekuensi Kerja RFID.....	11
4. Akurasi RFID	12
5. <i>Festures</i>	13
6. Definisi Pin.....	14
B. Transistor	14
C. IC Regulator Tegangan LM 7805	15
D. Mikrokontroler AVR (<i>Alf Vegard's Risch Processor</i>)	16
E. Mikrokontroler 328	18
1. Pengertian Mikrokontroler ATmega328.....	18
2. Daya	24
3. Memori.....	25
4. Komunikasi	25
F. Solenoid	26
1. Sistem Kerja Solenoid	26
G. Program	27
BAB III KONSEP RANCANGAN	29
A. Identifikasi Kebutuhan	29
B. Analisis Kebutuhan	29
C. <i>Flow Chart</i>	30
D. Pembuatan Alat	33
1. Perancangan perangkat keras (<i>hardware</i>).....	33
2. Perancangan rangkaian	34
3. Langkah perangkaian alat	39
4. Perancangan <i>software</i>	42
E. Pengujian Alat	47
F. Pengoperasian Alat	48
BAB IV HASIL PENGUJIAN DAN PEMBAHASAN	59
A. Pengujian	59
1. Pengujian Sensor RFID	50
2. Pengujian Material Bahan Penghalang.....	51
3. Pengujian Tegangan.....	52

4. Pengujian Solenoid	54
5. Pengujian Keseluruhan	55
B. Pembahasan	55
1. Sensor RFID	55
2. <i>Tag Card</i>	56
3. Tegangan	56
4. Solenoid	56
C. Unjuk Kerja	57
BAB V KESIMPULAN DAN SARAN	58
A. Kesimpulan	58
B. Keterbatasan Alat	59
C. Saran	59

DAFTAR TABEL

	halaman
Tabel 1. Spesifikasi RFID 125 KHz Modul.....	13
Tabel 2. Pengujian Jarak Sensor Dengan <i>Tag Card</i>	50
Tabel 3. Material Bahan Penghalang Yang Dapat Ditembus Oleh RFID <i>Reader</i> ...	51
Tabel 4. Hasil Pengukuran Tegangan	54
Tabel 5. Pengujian Solenoid	54
Tabel 6. Pengujian Keseluruhan	55

DAFTAR GAMBAR

	Halaman
Gambar 1. Cara Kerja RFID	9
Gambar 2. RFID 125 KHz Buatan <i>Itead Studio</i>	13
Gambar 3. Simbol tipe transistor	15
Gambar 4. Gambar PIN diagram IC LM 7805.....	15
Gambar 5. Konfigurasi Pin ATmega328	18
Gambar 6. Blok diagram ATmega328.....	21
Gambar 7. Status <i>Register</i> ATmega328.....	22
Gambar 8. Solenoid	26
Gambar 9. Cara Kerja Solenoid.....	27
Gambar 10. Pergerakan Solenoid	27
Gambar 11. <i>Flow Chart</i> Cara/Alur Kerja Pengaman Pintu Rumah Menggunakan RFID (<i>Radio Frequency Identification</i>) 125 KHz	31
Gambar 12. Diagram Blok	33
Gambar 13. Rangkaian Skematik Keseluruhan.....	35
Gambar 14. Blok Penurun Tegangan	34
Gambar 15. Rangkaian Sistem Minimum ATmega328 28 Pin	36
Gambar 16. Pengaturan Pin <i>Input Output</i> Sistem Mikrokontroler ATmega328	37
Gambar 17. Rangkaian <i>Driver</i> Solenoid	38
Gambar 18. Bentuk Fisik TIP122.....	38
Gambar 19. <i>Mechanical Data</i> TIP122	38
Gambar 20. Bentuk Fisik Kaki-Kaki RFID RDM630.....	39
Gambar 21. <i>Layout</i> Mikrokontroler ATmega328	41
Gambar 22. Tampak Atas Mikrokontroler ATmega328.....	41
Gambar 23. Tampak Bawah Mikrokontroler ATmega328.....	41
Gambar 24. <i>Layout PCB Driver</i> Solenoid	42
Gambar 25. Tampilan Jendela Program IDE Arduino 023	43
Gambar 26. Tampilan <i>Tools</i> Program <i>Port Serial</i> IDE Arduino 023.....	43

Gambar 27. Susunan Pin RFID Saat Pengukuran	49
Gambar 28. <i>Tag</i> Model Gantungan Kunci.....	51
Gambar 29. <i>Tag</i> Model <i>Card</i> (Kartu)	51
Gambar 30. Pengukuran Tegangan Solenoid	52
Gambar 31. Pengukuran Tegangan Keluaran Mikrokontroler Pada Pin Digital 13 Sebagai Pemicu Solenoid	52
Gambar 32. Pengukuran Tegangan Pada Modul RFID.	53
Gambar 33. Pengukuran Tegangan Pada Keluaran Regulator 7805	53

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Rangkaian Keseluruhan	62
Lampiran 2. <i>Layout</i> Dan Pcb Rangkaian Mikrokontroler ATmega328	63
Lampiran 3. <i>Layout</i> Dan Pcb Rangkaian <i>Driver</i> Solenoid	64
Lampiran 4. Daftar komponen	65
Lampiran 5. Program	67
Lampiran 6. Data Sheet RDM630 RFID Modul	70
Lampiran 7. Data Sheet TIP122	74
Lampiran 8. Data Sheet C7805	79
Lampiran 9. Data Sheet ATmega328	81

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Kurangnya tingkat keamanan dan mahalnya biaya pengamanan ekstra menjadi seringnya terjadi pencurian dan pembobolal pada rumah. Terkadang dirasa yakin ketika sedang ditinggal ataupun saat santai dirumah, bahkan diyakin pintu dalam keadaan terkunci. Namun pada kenyataan kasus pembobolan rumah pada zaman sekarang dengan mudahnya para pencuri membuka pengunci pada pintu yang terpasang hanya dengan seutas kawat atau pun dengan kunci tiruan lainnya. Keahlian para pencuri semakin hebat, oleh karena itu harus dipikirkan bagaimana caranya agar rumah tetap terjaga dan bebas dari para pencuri atau pembobol.

Seseorang mencuri di rumah dengan alasan bahwa rumah dilihat begitu mewah atau para pencuri tersebut mengetahui bahwa didalam rumah ada barang berharga. Tentu saja pemilik sadar akan hal tersebut dan segera mengamankan rumah dengan memberikan kunci ekstra pada pintu-pintu rumah. Namun kenapa tetap saja para pencuri masih dapat melakukan aksinya? Tentu saja jawabannya karena para pencuri memiliki kesempatan dan memiliki kemampuan untuk membuka pintu rumah tersebut.

Tidak dapat disangkal bahwa manusia tidak ada yang teliti 100%, sehingga kesempatan untuk para pencuri melakukan aksinya masih tetap ada. Banyak cara yang bisa dilakukan diantaranya dengan memasang kunci ekstra yang jumlahnya banyak sehingga membuat pencuri enggan

membobol rumah, namun apakah dirasa nyaman ketika ingin masuk ke dalam rumah dengan harus membuka satu persatu kunci ekstra sebelumnya dengan harus memenuhi saku yang mengganggu kenyamanan saat berpergian.

Perkembangan dunia kriminalitas dengan keahlian para pencuri yang semakin tinggi maka munculah gagasan inovasi alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz berbasis mikrokontroler ATmega328 yang tentunya dengan pengamanan yang tinggi. Rancangan keamanan ini tidak mengandalkan mekanik sebagai *interface*-nya melainkan menggunakan perangkat elektronik yang cukup sulit untuk dibobol karena selain diperlukan pengetahuan mengenai elektronik mereka juga harus memiliki pengetahuan dibidang pemrograman dan teknologi informasi yang tentunya juga tidak akan merepotkan kita dengan banyak kunci.

Berbeda dengan kunci mekanik, kunci elektronik pada rancangan keamanan ini menggunakan RFID sebagai pembukanya. Sistem RFID ini terdiri atas tiga komponen utama, yaitu *tag* atau *transponder*, *reader*, dan *database*. *Tag* RFID berfungsi sebagai alat pelabelan suatu objek yang di dalamnya terdapat sebuah data tentang objek tersebut. Kemudian *reader* RFID digunakan sebagai alat *scanning* atau pembaca informasi yang ada pada *tag* RFID tersebut. Sedangkan *database* digunakan sebagai pelacak dan penyimpan informasi tentang objek-objek yang dimiliki oleh *tag* RFID. RFID merupakan sebuah pengembangan dari sistem identifikasi

sebelumnya, yaitu *Barcode*. Perbedaan yang mendasar antara RFID dengan *barcode* terletak pada cara *scanning*, yaitu cara pembacaan sebuah *transponder* atau alat yang digunakan sebagai pelabelan. Untuk *barcode*, biasanya *scanning* dilakukan secara langsung dan posisi antara *tag* dengan *reader* harus benar. Jika tidak maka *tag* tersebut tidak dapat terbaca oleh *reader*. Berbeda dengan RFID yang hanya dengan mendekatkan *tag* ke *reader*, maka *tag* tersebut dapat teridentifikasi. Perangkat pengolah data berfungsi untuk mengolah data masukan yang akan diproses sebagai inputan identifikasi. Perangkat pengolah data terdiri dari mikrokontroler ATmega328. Perangkat penggerak berupa solenoid sebagai pengunci pengaman rumah. Perangkat identifikasi terdiri dari RFID *reader* beserta antena dan *tag* kartunya.

Reader yang di pasang dalam pintu, menambah keamanan karena tidak bias terlihat dari luar. Jika ingin menngganti kunci tidak perlu membongkar penguncinya seperti yang model dulu seperti grog, namun hanya mengganti syntag program yang tertanam pada RFID dan *reader*-nya tanpa harus membongkar. Selain itu penggunaan RFID ini juga dapat meminimalis keseluruhan kunci pada rumah, sehingga setiap anggota keluarga cukup membutuhkan satu *tag card* untuk membuka seluruh pintu yang ada di rumah.

B. Identifikasi Masalah

Berdasarkan pada latar belakang masalah yang ada, maka dapat diidentifikasi hal sebagai berikut :

1. Mudahnya para pencuri membuka pengunci pada pintu yang terpasang hanya dengan seutas kawat atau pun dengan kunci tiruan lainnya.
2. Tidak nyamanya saat harus membawa banyak kunci.
3. Kurangnya tingkat keamanan dan mahalnya biaya pengamanan ekstra menjadi seringnya terjadi pencurian dan pembobolal pada rumah.
4. Belum adanya pemanfaatan teknologi RFID dalam bidang keamanan.

C. Batasan Masalah

Berdasarkan pada pokok permasalahan yang diuraikan pada identifikasi masalah sebelumnya, maka batasan masalah pada proyek akhir ini adalah :

Alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328 ini hanya dibatasi pada pengujian unjuk kerja komunikasi teknologi RFID sebagai sarana identifikasi keamanan pintu dengan rancang bangun yang disesuaikan.

D. Rumusan Masalah

1. Bagaimana merancang *hardware*, alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328 ?
2. Bagaiman merealisasikan *software*, alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328 ?
3. Bagaimana unjuk kerja alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328 ?

E. Tujuan

Tujuan dari proyek akhir ini adalah :

1. Merealisasikan *hardware* pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328.
2. Mengaplikasikan *software* pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328.
3. Mengetahui unjuk kerja dari alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328.

F. Manfaat

Pembuatan proyek akhir ini diharapkan dapat bermanfaat bagi mahasiswa, lembaga pendidikan, dan industri. Adapun manfaat yang diharapkan dari pembuatan tugas akhir ini antara lain :

1. Bagi mahasiswa
 - a. Sebagai tolak ukur individual setelah mendapatkan ilmu dari bangku kuliah dan kehidupan sehari – hari untuk diimplementasikan dalam bentuk suatu alat.
 - b. Untuk mengaplikasikan ilmu yang didapat selama di bangku kuliah dan menerapkan ilmunya secara nyata.
 - c. Dapat digunakan sebagai bahan referensi atau pembelajaran dan penambah wawasan tentang alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz berbasis mikrokontroler khususnya ATmega 328 serta sebagai keajian untuk pengembangan selanjutnya.
 - d. Sebagai bentuk kontribusi terhadap Universitas dan pengabdian kepada masyarakat dalam bentuk karya alat yang bermanfaat.
2. Bagi program studi Teknik Elektronika
 - a. Sebagai wujud dari perkembangan Ilmu Pengetahuan dan Teknologi (IPTEK).
 - b. Sebagai parameter kualitas dan kuantitas lulusan mahasiswa Fakultas Teknik Universitas Negeri Yogyakarta.

3. Bagi Dunia Industri

Dapat digunakan sebagai pengembangan produk elektronika yang dapat diaplikasikan pada berbagai bidang khususnya pada bidang keamanan menggunakan RFID.

G. Keaslian Gagasan

Proyek Akhir dengan judul Alat Pengaman Pintu Rumah Menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328 terinspirasi dari teknologi RFID pada perancangan sistem absensi dengan RFID menggunakan *custom* RFID *reader* (Muhammad, Levy, 2009) yang menggunakan bahasa pemrograman java untuk *software*nya dan IC AT89S52. Bahasa pemrograman java dimanfaatkan untuk membuat tampilan *interface* alat yang didukung pula dengan bahasa pemrograman MySQL untuk *database*, *Hardware* pada alat ini menggunakan RFID model ID-12.

Adapun perbedaan penggunaan RFID pada perancangan sistem absensi dengan tugas akhir ini dari segi *hardware* menggunakan RFID model RDM-6300 yang berperan sebagai RFID *reader* beserta IC ATmega328 yang berfungsi sebagai pengolah data. Dari segi *software* tugas akhir ini menggunakan bahasa C model pemrograman Arduino yang memiliki *library* pemrograman yang luas dan juga menjadi *database* yang menyimpan data ID RFID *Card* yang bersifat valid.

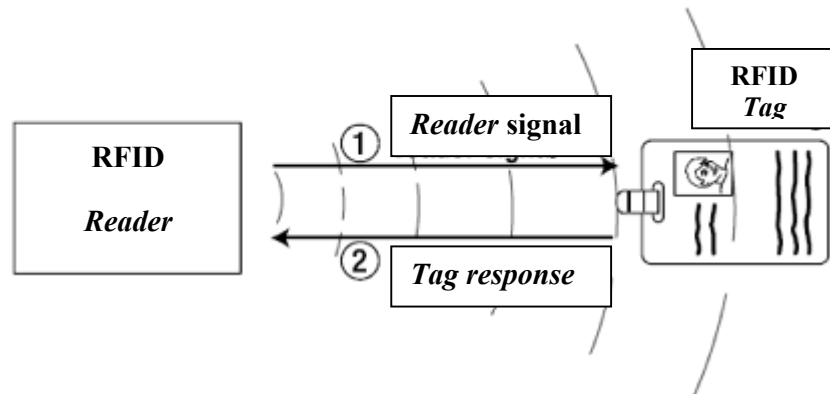
BAB II

PENDEKATAN PEMECAHAN MASALAH

A. RFID (*Radio Frequency Identification*)

RFID (*Radio Frequency Identification*) adalah teknologi identifikasi berbasis gelombang radio. Teknologi ini mampu mengidentifikasi berbagai objek secara simultan tanpa diperlukan kontak langsung (atau dalam jarak pendek). RFID dikembangkan sebagai pengganti atau penerus teknologi *barcode*. RFID bekerja pada HF (*High Frekuensi*) untuk aplikasi jarak dekat (*proximity*) dan bekerja pada UHF (*Ultra High Frekuensi*) untuk aplikasi jarak jauh (*vicinity*).

Sensor RFID adalah sensor yang mengidentifikasi suatu barang dengan menggunakan frekuensi radio. Sensor ini terdiri dari dua bagian penting: *transceiver (reader)* dan transponder (*tag*). Setiap *tag* tersimpan data yang berbeda. Data tersebut merupakan data identitas *tag*. *Reader* akan membaca data dari *tag* dengan perantara gelombang radio. Pada *reader* biasanya berhubungan dengan suatu mikrokontroler. Mikrokontroler ini berfungsi untuk mengolah data yang didapat *reader*. Struktur cara kerja RFID terdapat pada gambar 1.



Gambar 1. Cara Kerja RFID
(*Practical Arduino Cool Projects for Open Source Hardware*)

1. Pembaca RFID

Sebuah pembaca RFID harus menyelesaikan dua buah tugas, yaitu:

- a. Menerima perintah dari *software* aplikasi
- b. Berkomunikasi dengan *tag* RFID

Pembaca RFID adalah merupakan penghubung antara *software* aplikasi dengan antena yang akan meradiasikan gelombang radio ke *tag* RFID. Gelombang radio yang diemisikan oleh antena berpropagasi pada ruangan di sekitarnya. Akibatnya data dapat berpindah secara *wireless* ke *tag* RFID yang berada berdekatan dengan antena.

2. Tag RFID

Tag RFID adalah perangkat yang dibuat dari rangkaian elektronika dan antena yang terintegrasi di dalam rangkaian tersebut. Rangkaian elektronik dari *tag* RFID umumnya memiliki memori sehingga *tag* ini mempunyai kemampuan untuk menyimpan data. Memori pada *tag* secara dibagi menjadi sel-sel. Beberapa sel menyimpan data *Read Only*,

misalnya *serial number* yang unik yang disimpan pada saat *tag* tersebut diproduksi. Selain pada RFID mungkin juga dapat ditulis dan dibaca secara berulang.

Berdasarkan cara daya *tag*, *tag* RFID dapat digolongkan menjadi:

- a. *Tag* Aktif: yaitu *tag* yang daya diperoleh dari baterai, sehingga akan mengurangi daya yang diperlukan oleh pembaca RFID dan *tag* dapat mengirimkan informasi dalam jarak yang lebih jauh. Kelemahan dari tipe *tag* ini adalah harganya yang mahal dan ukurannya yang lebih besar karena lebih kompleks. Semakin banyak fungsi yang dapat dilakukan oleh *tag* RFID maka rangkaianannya akan semakin kompleks dan ukurannya akan semakin besar.
- b. *Tag* Pasif: yaitu *tag* yang daya diperoleh dari medan yang dihasilkan oleh pembaca RFID. Rangkaianannya lebih sederhana, harganya jauh lebih murah, ukurannya kecil, dan lebih ringan. Kelemahannya adalah *tag* hanya dapat mengirimkan informasi dalam jarak yang dekat dan pembaca RFID harus menyediakan daya tambahan untuk *tag* RFID.

Tag RFID telah sering dipertimbangkan untuk digunakan sebagai *barcode* pada masa yang akan datang. Pembacaan informasi pada *tag* RFID tidak memerlukan kontak sama sekali. Karena kemampuan rangkaian terintegrasi yang modern, maka *tag* RFID dapat menyimpan jauh lebih banyak informasi dibandingkan dengan *barcode*.

3. Frekuensi Kerja RFID

Faktor penting yang harus diperhatikan dalam RFID adalah frekuensi kerja dari sistem RFID. Ini adalah frekuensi yang digunakan untuk komunikasi *wireless* antara pembaca RFID dengan *tag* RFID.

Ada beberapa band frekuensi yang digunakan untuk sistem RFID yaitu:

<i>Low Frequency</i> (LF)	: 125 - 134 KHz
<i>High Frequency</i> (HF)	: 13.56 MHz
<i>Ultra High Frequency</i> (UHF)	: 868 - 956 MHz
<i>Microwave</i>	: 2.45 GHz

Pemilihan dari frekuensi kerja sistem RFID akan mempengaruhi jarak komunikasi, interferensi dengan frekuensi sistem radio lain, kecepatan komunikasi data, dan ukuran antena. Untuk frekuensi yang rendah (*Low Frequency* (LF) : 125 - 134 KHz) umumnya digunakan *tag* pasif (tidak memiliki sumber energi sendiri tanpa *battery*, Modulasi akan aktif setelah *tag* menerima gelombang elektromagnetik dari *reader*) dan untuk frekuensi tinggi (*High Frequency* (HF) : 13.56 MHz - *Microwave* : 2.45 GHz) digunakan *tag* aktif (memiliki sumber energi sendiri, modulasi aktif langsung dari *tag* sendiri). Pada frekuensi rendah, *tag* pasif tidak dapat mentransmisikan data dengan jarak yang jauh, karena keterbatasan daya yang diperoleh dari medan elektromagnetik. Akan tetapi komunikasi tetap dapat dilakukan tanpa kontak langsung. Pada penelitian ini hal yang perlu

mendapatkan perhatian adalah *tag* pasif harus terletak jauh dari objek logam, karena logam secara signifikan mengurangi fluks dari medan magnet. Akibatnya *tag* RFID tidak bekerja dengan baik, karena *tag* tidak menerima daya minimum untuk dapat bekerja.

Pada frekuensi tinggi, jarak komunikasi antara *tag* aktif dengan pembaca RFID dapat lebih jauh, tetapi masih terbatas oleh daya yang ada. Sinyal elektromagnetik pada frekuensi tinggi juga mendapatkan pelemahan (atenuasi) ketika *tag* tertutupi oleh es atau air. Pada kondisi terburuk, *tag* yang tertutup oleh logam tidak terdeteksi oleh pembaca RFID.

Ukuran antena yang harus digunakan untuk transmisi data bergantung dari panjang gelombang elektromagnetik. Untuk frekuensi yang rendah, maka antena harus dibuat dengan ukuran yang lebih besar dibandingkan dengan RFID dengan frekuensi tinggi.

4. Akurasi RFID

Akurasi RFID dapat didefinisikan sebagai tingkat keberhasilan pembaca RFID melakukan identifikasi sebuah *tag* yang berada pada area kerjanya. Keberhasilan dari proses identifikasi sangat dipengaruhi oleh beberapa batasan fisik, yaitu:

- a. Posisi antena pada pembaca RFID
- b. Karakteristik dari material lingkungan yang mencakup sistem RFID
- c. Batasan *catu* daya
- d. Frekuensi kerja sistem RFID

Sensor RFID yang digunakan adalah RFID 125KHz. Sensor ini digunakan karena RFID dengan frekuensi 125 KHz merupakan model yang banyak dijumpai dipasaran dan memiliki harga relatif terjangkau.



Gambar 2. RFID 125 Khz Buatan Itead Studio
(Iteadstudio <http://goo.gl/SYXmV>)

5. Features

- a. Support external antenna
- b. Maximum effective distance up to 150mm
- c. Less than 100ms decoding time
- d. UART TTL interface
- e. Support EM4100 compatible read only or read/write tags
- f. Built-in external bi-color LED and buzzer driver
- g. Small outline design

Tabel 1. Spesifikasi RFID 125 KHz Modul

Frequency	125KHz
Baud Rate	9600 (TTL Electricity Level RS232 format)
Interface	Weigang26 Or TTL Electricity Level RS232 format
Power supply	DC 5V ($\pm 5\%$)
Current	<50Ma
Operating range	>50mm(Depend on Card/Tag shape, manufacturer)
Expand I/O port	N/A
Indication light	N/A
Working temperature	-10°C~ +70°C
Storage temperature	-20°C~ +80°C
Max. humidity	Relative humidity 0 ~ 95%
Size	38.5mm × 19mm × 9mm

(Datasheet RFID 125 KHz Module)

6. Definisi Pin

Pin Definition (TTL interface RS232 data format):

P1:

PIN1 TX
 PIN2 RX
 PIN3
 PIN4 GND
 PIN5 +5V(DC)

P2:

PIN1 ANT1
 PIN2 ANT2

P3:

PIN1 LED
 PIN2 +5V(DC)
 PIN3 GND

B. Transistor

Transistor adalah komponen elektronika yang mempunyai tiga buah terminal. Terminal itu disebut emitor, basis, dan kolektor. Transistor seakan-akan dibentuk dari penggabungan dua buah dioda. Dioda satu dengan yang lain saling digabungkan dengan cara menyambungkan salah satu sisi dioda yang senama. Dengan cara penggabungan seperti ini dapat diperoleh satu buah transistor.

Transistor mempunyai 3 kaki. Anak panah yang terdapat di dalam simbol menunjukkan arah arus yang melalui transistor. Simbol tipe transistor dapat dilihat pada gambar 5:



Gambar 3. Simbol tipe transistor
(repository.usu.ac.id/bitstream/123456789/18326/.../Chapter%20II.pdf)

Keterangan :

C = kolektor

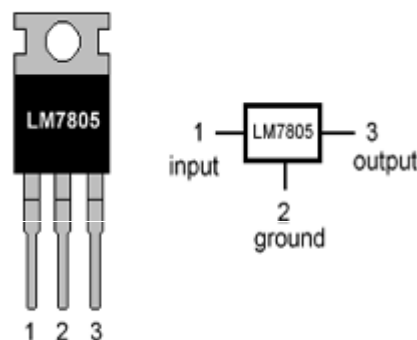
E = *emitter*

B = basis

C. IC Regulator Tegangan LM 7805

IC LM7805 adalah IC penyetabil tegangan 5 Volt DC yang memiliki kemampuan arus keluaran sampai 1 Ampere. Pada kemasan IC ini terdapat tiga kaki yaitu kaki pertama sebagai *input*, kaki kedua (tengah) sebagai kaki *ground* dan kaki ketiga sebagai *output* atau tegangan stabil 5 Volt.

LM7805 PINOUT DIAGRAM



Gambar 4. Gambar PIN diagram IC LM 7805
(<http://powersupplycircuit.net/lm7805.html>)

Pada badan kemasan IC ini terdapat besi yang berfungsi sebagai pendingin karena tegangan atau arus yang dikeluarkan oleh IC ini sangat dipengaruhi perubahan suhu komponen IC ini.

D. Mikrokontroler AVR (*Alf and Vegard's Risc Processor*)

Mikrokontroler adalah suatu alat, komponen pengontrol atau pengendali yang berukuran kecil (mikro). Mikrokontroler merupakan komputer di dalam chip yang digunakan untuk mengontrol peralatan elektronik, yang menekankan efisiensi dan efektifitas biaya. Secara harfiah bisa disebut pengendali kecil sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti IC TTL dan CMOS dapat direduksi/diperkecil dan akhirnya terpusat serta dikendalikan oleh mikrokontroler.

Secara teknis hanya ada 2 jenis mikrokontroler yaitu RISC dan CISC. Masing-masing mempunyai keturunan/keluarga sendiri-sendiri. RISC kependekan dari *Reduced Instruction Set Computer* yang memiliki instruksi terbatas tapi memiliki fasilitas yang lebih banyak. CISC kependekan dari *Complex Instruction Set Computer* yang memiliki instruksi lebih lengkap tetapi dengan fasilitas secukupnya.

Mikrokontroler sudah mengandung beberapa peripheral yang langsung bisa dimanfaatkan, misalnya *port* paralel, *port* serial, komparator, konversi digital ke analog (DAC), konversi analog ke digital dan sebagainya hanya menggunakan sistem minimum yang tidak rumit atau kompleks. Bila

dibandingkan dengan mikroprosesor maka mikrokontroler jauh lebih unggul (LAMEL,2006).

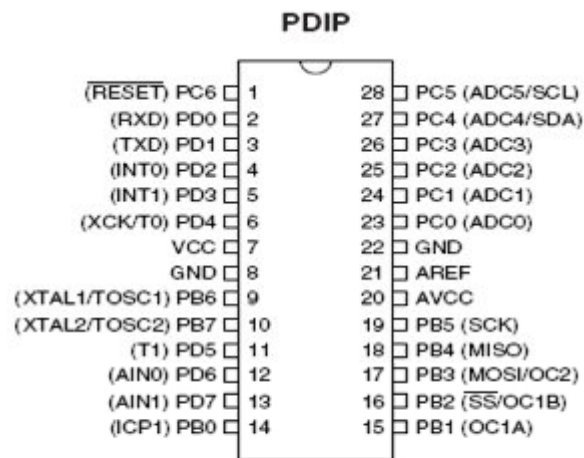
Beberapa produsen mikrokontroler yang dikenal yaitu Atmel, Microchip, Motorola, Rensas dan Phillips yang menciptakan mikrokontroler dengan kelebihan masing-masing. Instruksi-instruksi dari sebuah program pada tiap jenis mikrokontroller mempunyai beberapa perbedaan, misalnya instruksi pada mikrokontroller Atmel berbeda dengan instruksi mikrokontroller Motorola. Pada prinsipnya program pada mikrokontroller dijalankan secara bertahap (Suhata ST, 2004). Kurniawan (2006) mengemukakan bahwa ada beberapa faktor penting yang menjadi pertimbangan dalam memilih mikrokontroler yang akan digunakan diantaranya :

1. Harga mikrokontroler.
2. Ukuran memori mikrokontroler.
3. Fitur ADC, timer dan fasilitas komunikasi.
4. Fitur utama lain seperti pengontrol utama akuisi data, penampil LCD dan lainnya.
5. Kecepatan eksekusi intruksi.
6. Dukungan *softwear* yang dapat digunakan
7. Kebutuhan sistem yang akan digunakan.

E. Mikrokontroler ATmega328

1. Pengertian Mikrokontroler ATmega328

Mikrokontroler ATmega328 memiliki 14 *input* digital *output* pin/(6 *output* PWM), 6 *input* analog, 16 MHz osilator kristal, koneksi serial, ICSP *header*, dan tombol *reset*. Ini berisi semua fitur yang diperlukan untuk mendukung mikrokontroler, cukup hubungkan ke komputer dengan kabel USB *to Serial* atau listrik AC yang ke adaptor DC/baterai untuk memulai.



Gambar 5. Konfigurasi Pin ATmega328

(Datasheet ATmega328)

ATmega328 memiliki 28 pin yang masing-masing pin-nya memiliki fungsi yang berbeda-beda baik sebagai *port* ataupun sebagai fungsi yang lain. Berikut akan dijelaskan tentang kegunaan dari masing-masing kaki pada ATmega328.

1) VCC

Merupakan *supply* tegangan untuk digital.

2) GND

Merupakan *ground* untuk semua komponen yang membutuhkan *grounding*.

3) Port B

Di dalam *port B* terdapat XTAL1, XTAL2, TOSC1, TOSC2. Jumlah *port B* adalah 8 buah pin mulai dari pin B.0 sampai dengan pin B.7. Tiap pin dapat digunakan sebagai *input* dan juga *output*. *Port B* merupakan sebuah 8-bit *bi-directional I/O port* dengan internal *pull-up* resistor. Sebagai *input*, pin-pin yang terdapat pada *port B* yang secara eksternal diturunkan, maka akan mengeluarkan arus jika *pull-up* resistor diaktifkan. Jika ingin menggunakan tambahan kristal, maka cukup menghubungkan kaki dari kristal ke kaki pada pin *port B*. Namun jika tidak digunakan, maka cukup dibiarkan saja. Penggunaan kegunaan dari masing-masing kaki ditentukan dari *clock fuse setting*-nya.

4) Port C

Port C merupakan sebuah 7-bit *bi-directional I/O port* yang di dalam masing-masing pin terdapat *pull-up* resistor. Jumlah pin-nya hanya 7 buah mulai dari pin C.0 sampai dengan pin C.6. Sebagai keluaran / *output*, *port C* memiliki karakteristik yang sama dalam hal kemampuan menyerap arus (*sink*) ataupun mengeluarkan arus (*source*).

5) Reset / PC6

Jika RSTDISBL *Fuse* diprogram, maka PC6 akan berfungsi sebagai pin I/O. Untuk diperhatikan juga bahwa pin ini memiliki

karakteristik yang berbeda dengan pin-pin yang terdapat pada *port C*. Namun jika *RSTDISBL Fuse* tidak diprogram, maka pin ini akan berfungsi sebagai *input reset*. Dan jika level tegangan yang masuk ke pin ini rendah dan pulsa yang ada lebih pendek dari pulsa minimum, maka akan menghasilkan suatu kondisi *reset* meskipun *clock*-nya tidak bekerja.

6) *Port D*

Port D merupakan 8-bit bi-directional I/O dengan *internal pull-up* resistor. Fungsi dari *port* ini sama dengan *port-port* yang lain. Hanya saja pada *port* ini tidak terdapat kegunaan-kegunaan yang lain. Pada *port* ini hanya berfungsi sebagai masukan dan keluaran saja atau biasa disebut dengan I/O.

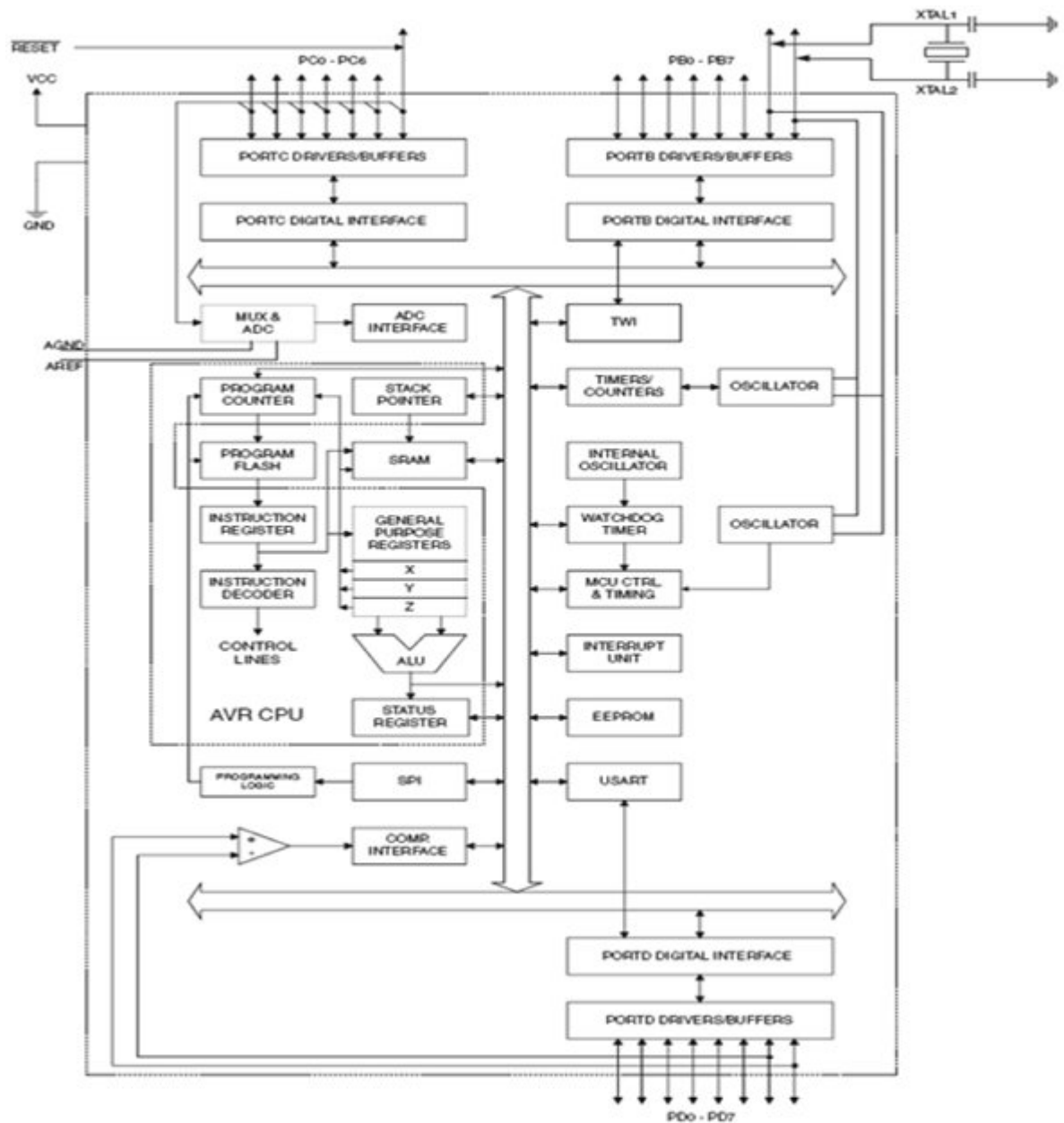
7) *AVCC*

Pada pin ini memiliki fungsi sebagai *supply* tegangan untuk ADC. Untuk pin ini harus dihubungkan secara terpisah dengan VCC karena pin ini digunakan untuk analog saja. Bahkan jika ADC pada AVR tidak digunakan, tetap saja disarankan untuk menghubungkan secara terpisah dengan VCC. Cara menghubungkan *AVCC* adalah melewati *low-pass filter* setelah itu dihubungkan dengan VCC.

8) *AREF*

Merupakan pin referensi analog jika menggunakan ADC.

Gambar blok diagram ATmega328 pada gambar 6 :

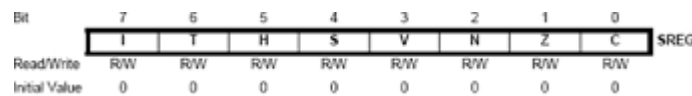


Gambar 6. Blok diagram ATmega328
(Datasheet ATmega328)

Pada AVR status register mengandung beberapa informasi mengenai hasil dari kebanyakan hasil eksekusi instruksi aritmatik. Informasi ini dapat digunakan untuk *altering* arus program sebagai kegunaan untuk

meningkatkan performa pengoperasian. Perlu diketahui bahwa register ini di-*update* setelah semua operasi ALU (*Arithmetic Logic Unit*). Hal tersebut seperti yang telah tertulis dalam *datasheet* khususnya pada bagian *Instruction Set Reference*.

Dalam hal ini untuk beberapa kasus dapat membuang kebutuhan penggunaan instruksi perbandingan yang telah didedikasikan serta dapat menghasilkan peningkatan dalam hal kecepatan dan kode yang lebih sederhana dan singkat. Register ini tidak secara otomatis tersimpan ketika memasuki sebuah rutin interupsi dan juga ketika menjalankan sebuah perintah setelah kembali dari interupsi. Namun hal tersebut harus dilakukan melalui *software*. Berikut adalah gambar status register.



Gambar 7. Status Register ATmega328
(*Datasheet ATmega328*)

Masing - masing *bit* yang terlihat di atas adalah berfungsi sebagai berikut :

9) Bit 7 (I)

Merupakan *bit Global Interrupt Enable*. Bit ini harus di-*set* supaya semua perintah interupsi dapat dijalankan. Untuk fungsi *interupsi* individual akan dijelaskan pada bagian yang lain. Jika *bit* ini di-*reset*, maka semua perintah interupsi baik yang individual maupun yang secara umum akan diabaikan. *Bit* ini akan dibersihkan atau *cleared* oleh *hardware* setelah sebuah interupsi dijalankan dan akan di-*set* kembali

oleh perintah RETI. *Bit* ini juga dapat di-*set* dan di-*reset* melalui aplikasi dengan instruksi SEI dan CLI.

10) *Bit 6 (T)*

Merupakan *bit Copy Storage*. Instruksi *bit Copy Instructions* BLD (*Bit Load*) dan BST (*Bit Store*) menggunakan *bit* ini sebagai asal atau tujuan untuk *bit* yang telah dioperasikan. Sebuah *bit* dari sebuah *register* dalam *register file* dapat disalin ke dalam *bit* ini dengan menggunakan instruksi BST, dan sebuah *bit* di dalam *bit* ini dapat disalin ke dalam sebuah *bit* di dalam *register* pada *register file* dengan menggunakan perintah BLD.

11) *Bit 5 (H)*

Merupakan *bit Half Carry Flag*. *Bit* ini menandakan sebuah *Half Carry* dalam beberapa operasi aritmatika. *Bit* ini berfungsi dalam aritmatik BCD.

12) *Bit 4 (S)*

Merupakan *Sign bit*. *Bit* ini selalu merupakan sebuah eksklusif di antara *Negative Flag (N)* dan *Two's Complement Overflow Flag (V)*. $S = N * V$.

13) *Bit 3 (V)*

Merupakan *bit Two's Complement Overflow Flag*. *Bit* ini menyediakan fungsi aritmatika dua komplemen.

14) *Bit 2 (N)*

Merupakan *bit Negative Flag*. *Bit* ini mengindikasikan sebuah hasil *negative* di dalam sebuah fungsi logika atau aritmatika.

15) *Bit 1 (Z)*

Merupakan *bit Zero Flag*. Bit ini mengindikasikan sebuah hasil nol “0” dalam sebuah fungsi aritmatika atau logika.

16) *Bit 0 (C)*

Merupakan *bit Carry Flag*. Bit ini mengindikasikan sebuah *carry* atau sisa dalam sebuah fungsi aritmatika atau logika.

2. Daya

Mikrokontroler ATmega328 dapat diaktifkan dengan catu daya eksternal. *Adaptor* ini dapat dihubungkan dengan menancapkan *plug* positif 2.1mm ke colokan listrik. Dari baterai dapat dimasukkan dalam *Gnd* dan *Vin* pin *header* dari *konektor POWER*. Mikrokontroler ATmega328 ini dapat beroperasi pada pasokan tegangan eksternal 6 sampai 20 volt. Jika diberikan dengan kurang dari 7V, pin yang keluaran 5V mungkin pasokannya kurang dari 5 volt dan mikrokontroler ATmega328 mungkin tidak stabil. Jika menggunakan lebih dari 12V, regulator tegangan bisa panas dan merusak *IC* mikro. Kisaran yang disarankan adalah 7 - 12 volt.

Pin sumber daya dalam mikrokontroler ATmega328 ini adalah sebagai berikut:

a) *VIN*

Tegangan masukan pada mikrokontroler ATmega328 menggunakan sumber daya eksternal.

b) 5V

Catu daya 5 volt ini digunakan untuk daya mikrokontroler dan komponen lainnya pada *board* mikrokontroler ATmega328. Hal ini dapat terjadi dilakukan dari pin *VIN* melalui regulator *on-board*, atau melalui *port* USB atau sumber tegangan lainnya seperti adaptor.

c) *GND*.

Pin *Ground*

3. Memori

Mikrokontroler ATmega328 memiliki 32 KB memori *flash* untuk menyimpan kode (sedangkan 2 KB digunakan untuk *bootloader*). Mikrokontroler ATmega328 memiliki 2 KB dari SRAM dan 1 KB dari EEPROM yang dapat dibaca dan ditulis.

4. Komunikasi

Mikrokontroler ATmega328 memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, atau mikrokontroler lainnya. Mikrokontroler ATmega328 memiliki UART TTL (5V) komunikasi serial, yang tersedia di pin digital 0 (RX) dan 1 (TX).

Sebuah *library Softwear Serial* memungkinkan untuk komunikasi serial pada salah satu pin mikrokontroler ATmega328 digital.

Mikrokontroler ATmega328 juga mendukung I2C (TWI) dan komunikasi SPI.

F. Solenoid

Solenoid adalah aktuator yang mampu melakukan gerakan linier. solenoid dapat elektromekanis (AC/DC), hidrolik, pneumatik atau didorong semua operasi pada prinsip-prinsip dasar yang sama. Dengan memberikan sumber tegangan maka solenoid dapat menghasilkan gaya yang linier. Contohnya untuk menekan tombol, memukul tombol pada piano, operator katup, dan bahkan untuk robot melompat. Solenoids DC beroperasi pada prinsip-prinsip seperti motor DC. Perbedaan antara solenoida dan motor adalah bahwa solenoid adalah motor yang tidak dapat berputar.

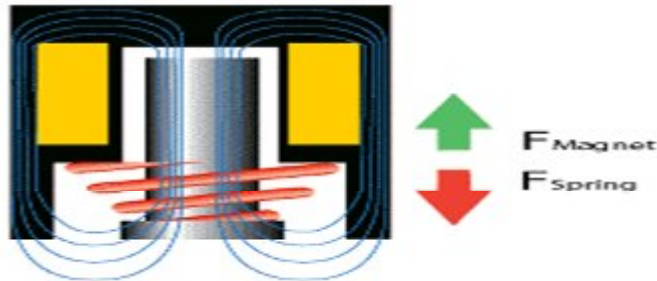


Gambar 8. Solenoid
(<http://goo.gl/2IcDX>)

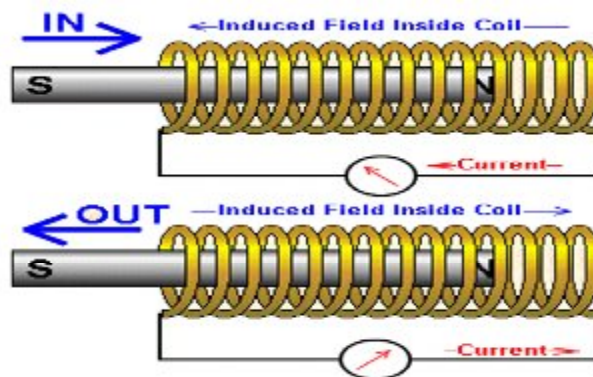
1. Sistem Kerja Solenoid

Di dalam solenoida terdapat kawat melingkar pada inti besi. Ketika arus listrik melalui kawat ini, maka terjadi medan magnet untuk menghasilkan energi yang bisa mendorong inti besi. Poros dalam dari solenoid adalah piston seperti silinder terbuat dari besi atau baja, yang

disebut *plunger* (setara dengan sebuah dinamo). Medan magnet kemudian menerapkan kekuatan untuk plunger ini, baik menarik atau *repeling* (kembali posisi). Ketika medan magnet dimatikan, pegas *plunger* kemudian kembali ke keadaan semula.



Gambar 9. Cara Kerja Solenoid
(<http://goo.gl/ppnIq>)



Gambar 10. Pergerakan Solenoid
(<http://goo.gl/1as0X>)

G. Pemrograman

Selain perangkat mikrokontroler dan RFID, desain pengaman pintu rumah ini juga memerlukan *softwear* untuk memprogram mikrokontroler. Adapun *softwear* yang digunakan adalah *integrated development environment* (IDE) Arduino 023. IDE Arduino ini didukung dengan *library* yang memudahkan penggunaanya dalam membuat program untuk

mikrokontroler. IDE Arduino 023 ini mampu berjalan di *multi platform*.

Kelebihan yang dimiliki IDE Arduino 023 antara lain:

1. Merupakan IDE (*Integrated Development Environment*)
2. Mendukung standard bahasa *C* dan *C++*
3. Memiliki dukungan *library* yang lengkap
4. Memiliki fasilitas untuk meng-*upload* program langsung dari IDE Arduino 023 tanpa menggunakan *hardware* tambahan
5. Memiliki fasilitas untuk menyalin kode program ke bahasa HTML
6. Memiliki fasilitas untuk menyalin kode program ke bahasa BB code
7. Mampu digunakan dengan dukungan *software* pihak kedua seperti Processing
8. Memiliki fasilitas serial monitor tersendiri yang terintegrasi di dalam IDE Arduino 023, sehingga dapat digunakan untuk membantu pengecekan program yang menggunakan fasilitas komunikasi serial.
9. Memiliki kemampuan *interfacing software* dengan *Python*, *Instant Reality (X3D)*, *Flash*, *Processing*, *PD (Pure Data)*, *MaxMSP*, *VVVV*, *Director*, *Ruby*, *C*, *Linux TTY*, *SuperCollider*, *Second Life*, *Squeak*, *Mathematica*, *C++*

(<http://arduino.cc/playground/Main/InterfacingWithSoftware>)

BAB III

KONSEP RANCANGAN

A. Identifikasi Kebutuhan

Dalam merancang alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz berbasis mikrokontroler ATmega328 dibutuhkan beberapa komponen yang terdiri atas:

1. Adaptor 12V untuk mensuplai tegangan agar bekerja dengan baik.
2. Sistem mikrokontroler ATmega328 sebagai sistem pengolah *INPUT/OUTPUT*.
3. Solenoid sebagai pengunci pintu.
4. Rangkaian *driver* solenoid untuk mengendalikan solenoid.
5. Modul RFID (*Radio Frequency Identification*) 125 KHz sebagai sarana penerapan teknologi RFID.
6. Rancang bangun pintu rumah sebagai prototipe proyek akhir.

B. Analisis Kebutuhan

Berdasarkan identifikasi kebutuhan yang ada, maka diperlukan beberapa spesifikasi dari komponen atau rangkaian sebagai berikut :

1. Perangkat *Power Supply*

Perangkat *Power Supply* yang digunakan berupa Adaptor 12V.

2. Perangkat Mikrokontroler

Perangkat kendali mikrokontroler pada alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz ini

menggunakan mikrokontroler ATmega328 sebagai perangkat mikrokontroler pengendalinya.

3. Perangkat Solenoid

Perangkat solenoid sebagai pengunci pintu menggunakan mini solenoid 5V yang merupakan kumparan yang menginduksi intibesi sehingga dapat berfungsi layaknya kerja magnet terhadap besi.

4. Rangkaian *driver* solenoid

Driver solenoid sebagai rangkaian yang *men-drive* kerja solenoid agar sesuai dengan *INPUT*-an yang ada. *Driver* adalah *interface* yang digunakan sebagai pengendali komponen yang memiliki tegangan kerja lebih besar atau lebih kecil dari tegangan kerja pengendalinya yang difungsikan sebagai sinyal *INPUT*-an.

5. Modul RFID

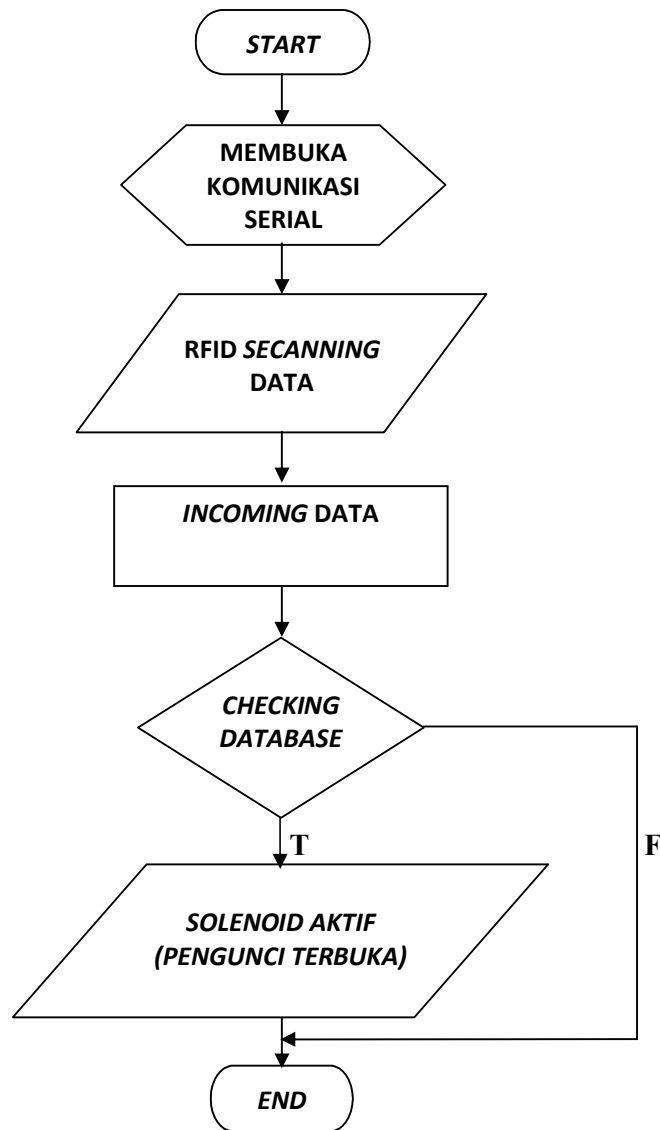
Modul RFID 125 KHz sebagai modul untuk membaca *tag card* yang kemudian diproses sebagai data masukan.

6. Rancang bangun pintu

Rancang bangun pintu digunakan sebagai model unjuk kerja alat.

C. *Flow Chart*

Berikut ini adalah gambar *flow chart* untuk alur cara kerja alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz berbasis mikrokontroler ATmega328 pada gambar 11.



Gambar 11. *Flow chart* cara kerja pengaman pintu rumah menggunakan RFID (radio frequency identification) 125 KHz

1. Penjelasan *Flowchart*

a. Membuka komunikasi serial

Pada saat alat diaktifkan dengan sistem RFID maka sistem mikrokontroler akan menjalankan komunikasi serial untuk jalan bagi data yang dibaca oleh RFID *reader* melalui komunikasi serial agar dapat di proses kembali oleh mikrokontroler.

b. RFID *Reader scanning* data

Sistem RFID yang meliputi RFID *reader* saat diaktifkan akan menjalankan fungsinya dalam men-*scanning* data yang masuk melalui *reader* (antenna). Data yang masuk akan diolah oleh mikrokontroler dan disesuaikan dengan *database* ID yang ada di dalam program.

c. *Incoming* Data

Data yang dibaca oleh RFID *reader* (Antena) akan masuk melalui sesi *Incoming* data ini yang nantinya akan diproses oleh mikrokontroler.

d. *Checking Database*

Setiap data yang berhasil dibaca akan di chek kesesuaiannya dengan data yang ada pada *database* program yang ada di dalam mikrokontroler.

e. Solenoid Aktif

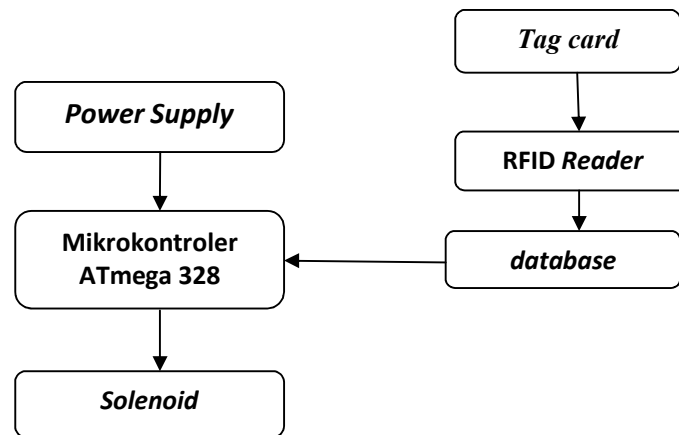
Apabila data yang dibaca sesuai dengan yang ada pada *database* mikrokontroler maka mikrokontroler akan mengeluarkan *output*

perintah agar solenoid dalam kondisi aktif yang membuat pengunci pintu terbuka akibat kondisi solenoid yang aktif.

D. Pembuatan Alat

1. Perancangan perangkat keras (*hardware*)

Rangkaian alat pengaman pintu rumah menggunakan RFID (*Radio Frequency Identification*) 125 KHz Berbasis Mikrokontroler ATmega328



Gambar 12. Diagram Blok

a. Cara kerja blok *INPUT*

Blok *INPUT* adalah *tag card* yang akan dibaca oleh RFID *reader* dan data yang terbaca akan dicek dan diproses oleh mikrokontroler apakah sesuai dengan *database* atau tidak.

b. Mikrokontroler ATmega328

Sistem kontrol yang digunakan adalah sistem mikrokontroler ATmega328 dengan rancang bangun yang disesuaikan agar sesuai dengan modul RFID yang digunakan.

c. Blok *OUTPUT*

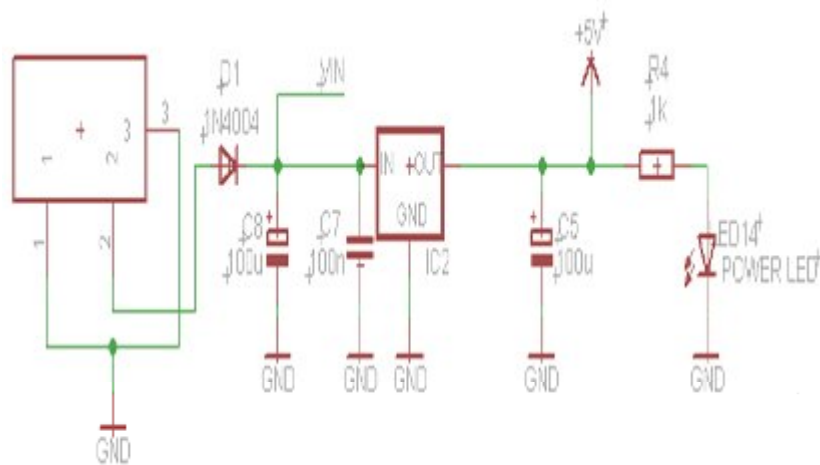
Pada sistem ini memfungsikan solenoid sebagai *OUTPUT* yang akan berperan sebagai pengunci pintu dengan memanfaatkan cara kerja induksi layaknya magnet.

2. Perencanaan Rangkaian

a. *Power Supply*

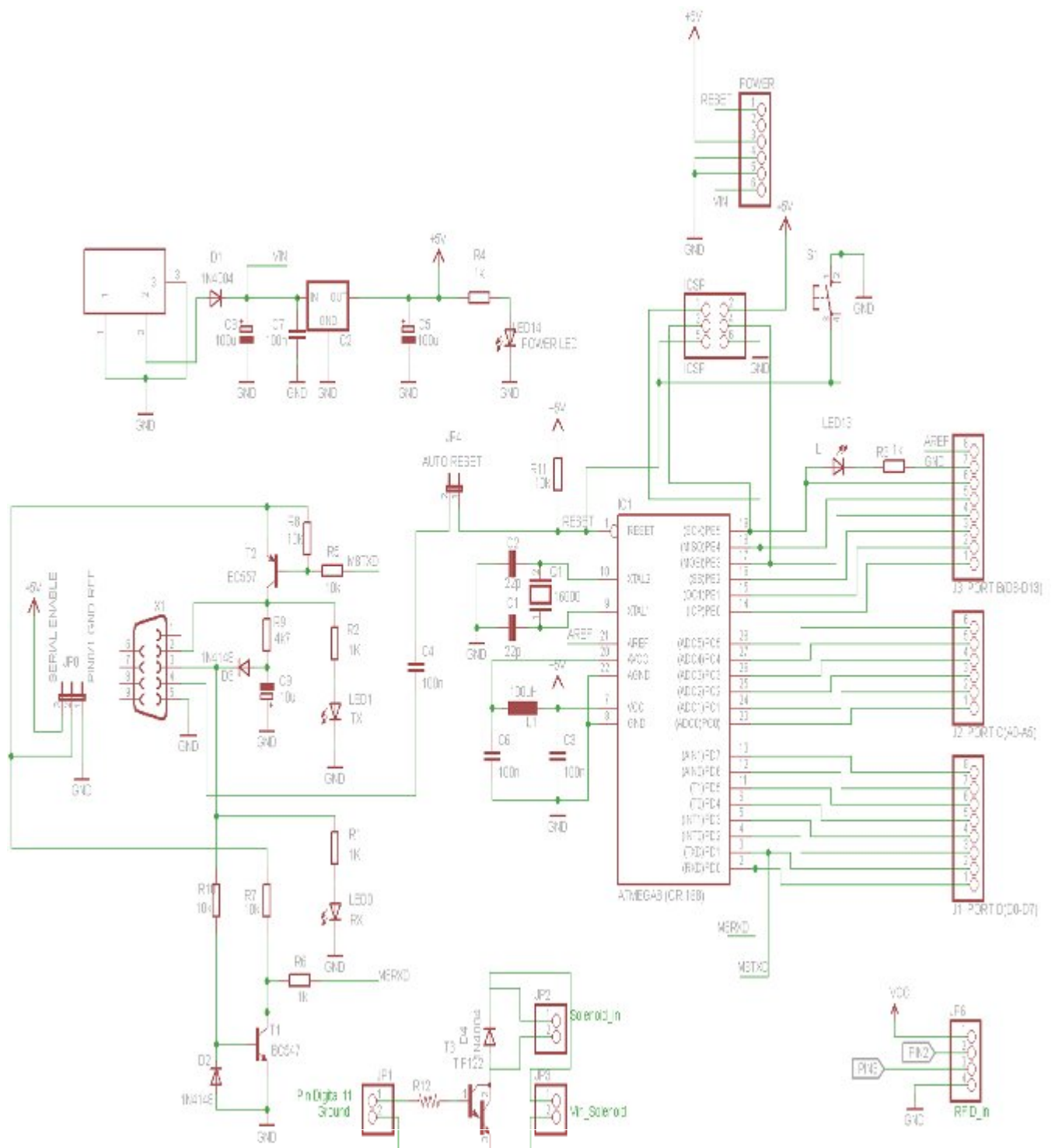
Power Supply sangat penting untuk menyuplai tegangan ke sistem mikrokontroler dan solenoid. *Power Supply* pada alat ini menggunakan Adaptor 12V.

b. Rangkaian Keseruruhan pada gambar 13.

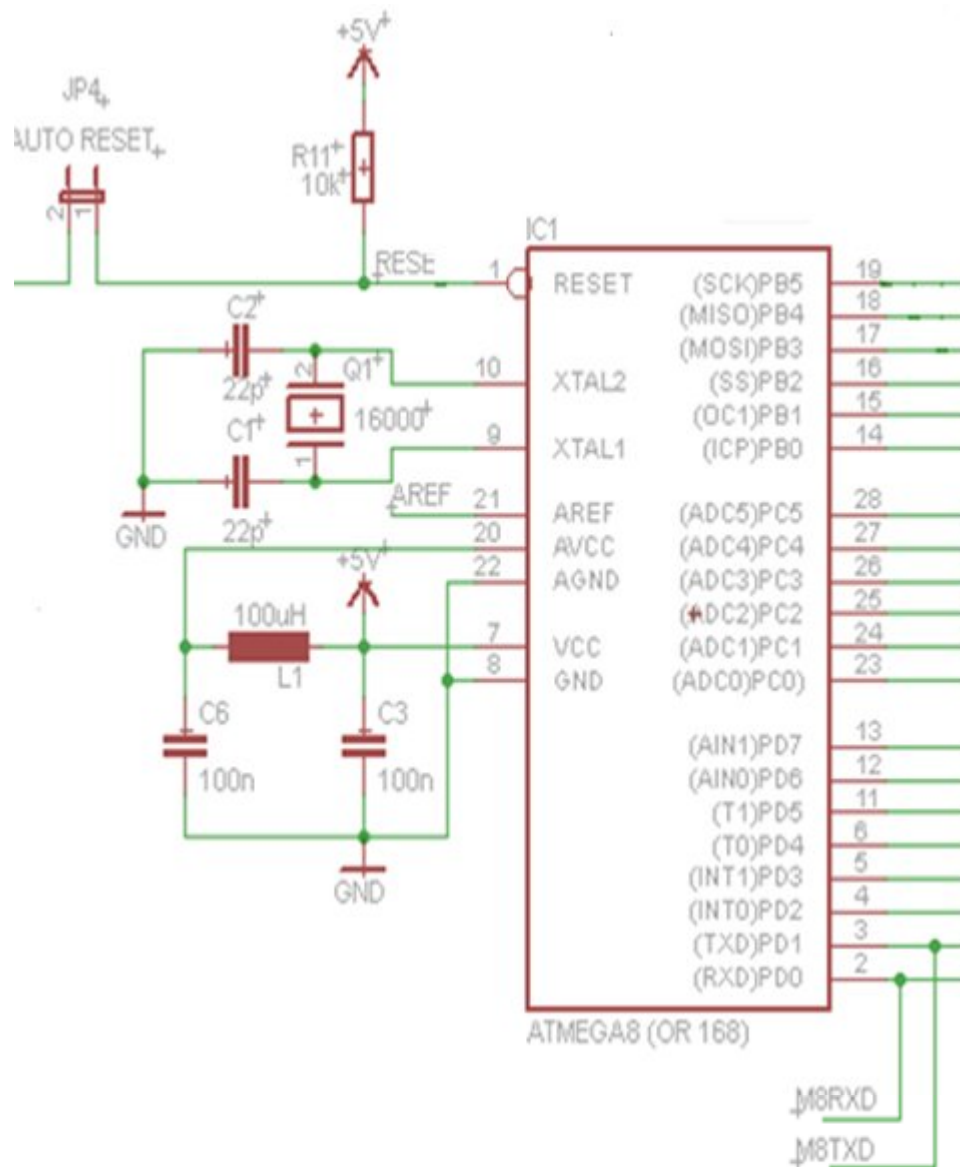


Gambar 14. Blok penurun tegangan

Gambar 14 merupakan bagian blok penurun tegangan dengan fungsi menurunkan tegangan yang masuk melalui *jack* dc menggunakan regulator 7805 yang akan menghasilkan tegangan 5 volt agar sesuai dengan kebutuhan suplai sistem mikrokontroler.

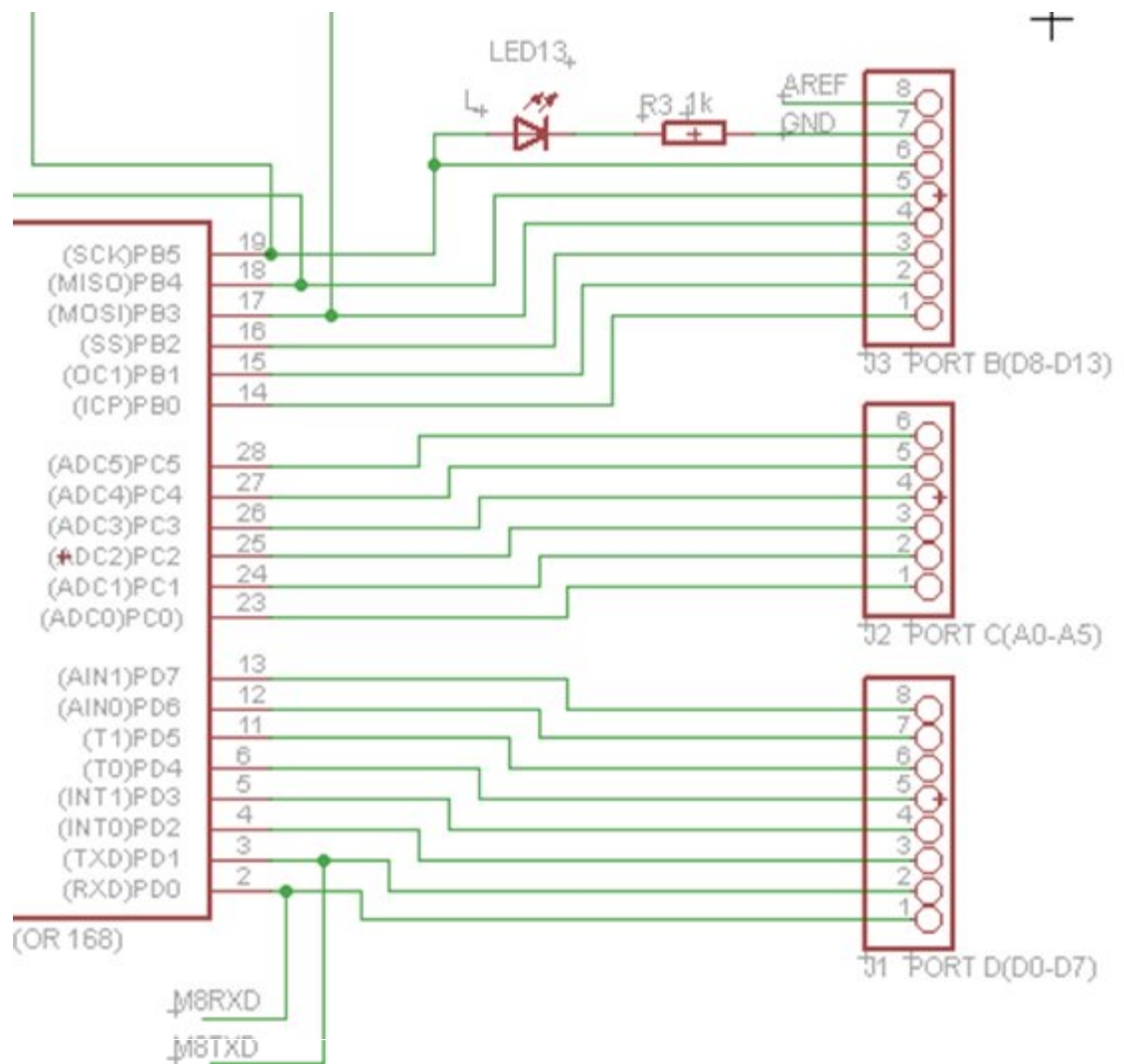


Gambar 13. Rangkaian skematik keseluruhan.



Gambar 15. Rangkaian sistem minimum ATmega328 28 Pin.

Gambar 15 merupakan bagian dari sistem minimum mikrokontroler ATmega328 yang berfungsi sebagai pengolah data seluruh proses sistem.

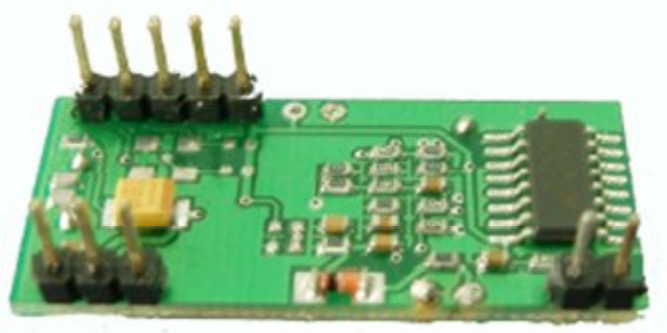


Gambar 16. Pengaturan Pin *INPUT OUTPUT* Sistem Mikrokontroler ATmega328.

Gambar 16 merupakan pengaturan letak pin dari IC ATmega328 yang terhubung pada pin *header* yang dapat digunakan sebagai masukan atau keluaran.

interface yang digunakan sebagai pengendali komponen yang memiliki tegangan kerja lebih besar atau lebih kecil dari tegangan kerja pengendalinya yang difungsikan sebagai sinyal *INPUT*-an.

d. Rangkaian RFID RDM6300



Gambar 20. Bentuk fisik kaki-kaki RFID RDM630
(*Datasheet RDM630*)

RFID RDM6300 ini bekerja pada frekuensi 125KHz dengan jarak pembacaan 4 cm. RFID RDM630 memerlukan suplai 5V agar dapat beroperasi dengan baik.

3. Langkah Perangkaian Alat (*hardware*)

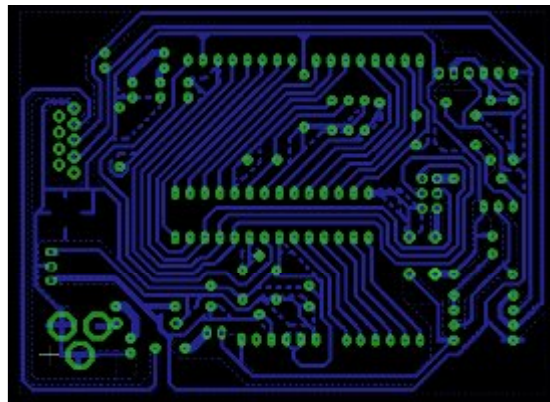
a. Perangkaian *Hardware*

1) Perangkaian rangkaian mikrokontroler ATmega328.

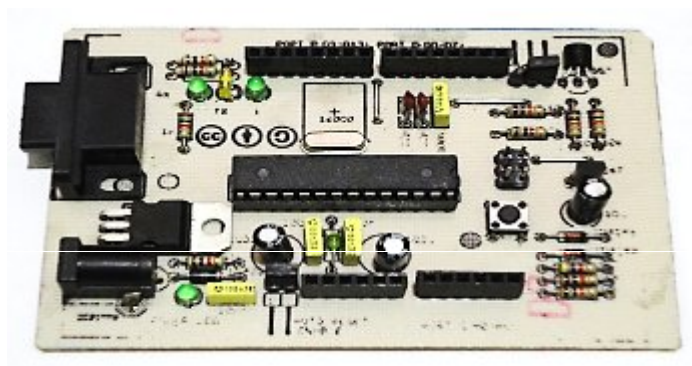
Bahan dan alat yang diperlukan antara lain :

- a) Perangkat komputer
- b) PCB Fiber polos
- c) FeCl_3
- d) Bor
- e) Solder
- f) Aklirilk
- g) Tenol

- 2) Proses perancangan rangkaian elektronik.
 - a) Merancang gambar skematik rangkaian
 - b) Mendesain *Layout* PCB
 - c) Mencetak desain PCB
 - d) Melarutkan PCB menggunakan larutan FeCl_3
 - e) Mengebor PCB sesuai *layout* desain PCB
 - f) Memasang dan menyolder komponen ke papan PCB
 - g) Melakukan pengujian
- 3) Komponen yang di perlukan dalam perancangan rangkaian mikrokontroler ATmega328 adalah sebagai berikut :
 - a) Capacitor Ceramic 22pF
 - b) *Capasitor Ceramic* 100nF
 - c) *Capasitor Elektrolic* 100 uF 16v
 - d) *Capasitor non polar elektrolic* 10uF 16v
 - e) Dioda 1N4004
 - f) Dioda 1N4148
 - g) *Jack* DC PCB
 - h) Regulator 7805
 - i) Induktor 100uH
 - j) Led Hijau 3mm
 - k) 16 MHz *Crystal*
 - l) Resistor 1k 1/4 Watt
 - m) Resistor 4k7 1/4 Watt
 - n) Resistor 10k 1/4 Watt
 - o) Transistor BC547 NPN
 - p) Transistor BC557 PNP
 - q) DB9 Pcb



Gambar 21. *Layout* Mikrokontroler ATmega328

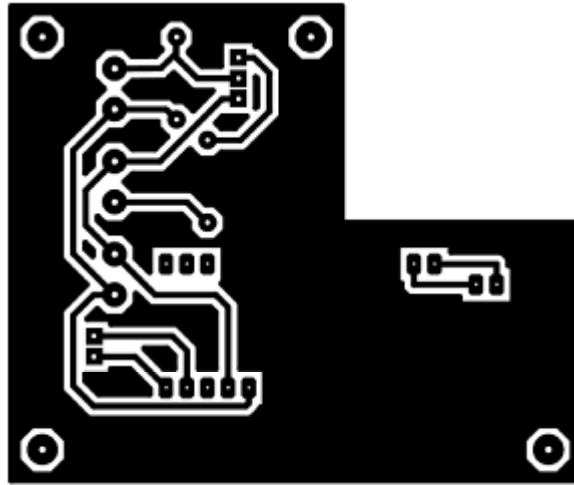


Gambar 22. Tampak Atas Mikrokontroler ATmega328



Gambar 23. Tampak Bawah Mikrokontroler ATmega328

b. *Layout PCB driver solenoid*



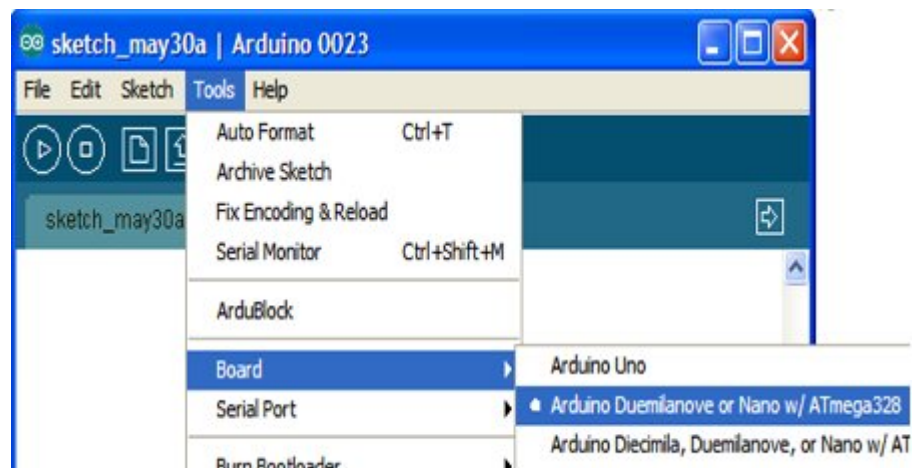
Gambar 24. *Layout PCB Driver solenoid*

Layout PCB Driver solenoid diperlukan untuk memudahkan dalam penggunaannya. *Layout* ini dibuat menggunakan *software* Proteus.

4. *Perancangan Software*

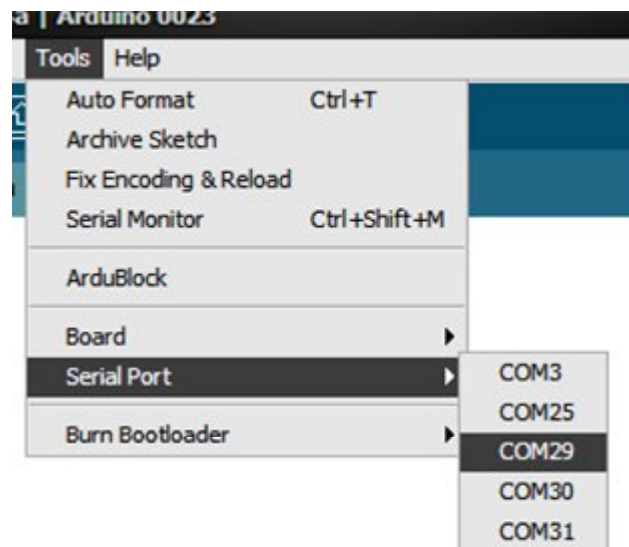
Aplikasi dan modifikasi *syntac perangkat lunak* pada mikrokontroler ATmega328 ini dimaksudkan untuk membuat *database* terprogram yang berisi serial ID dari *tag card* yang digunakan dan melakukan proses pengolahan data masukan menjadi perintah untuk proses penguncian oleh solenoid. Hal yang dilakukan berupa inisialisasi dan penulisan *listing* program.

a. Inisialisasi model mikrokontroler yang digunakan pada gambar 25:



Gambar 25. Tampilan Jendela Program IDE Arduino 023

b. Inisialisasi *port* serial



Gambar 26. Tampilan *tools* Program *port* serial IDE Arduino 023

Pada pembuatan *listing program* menggunakan IDE Arduino 023 yang menggunakan *basic* bahasa C.

c. Dasar – Dasar Program

1. *Void setup()*

Berisi kode program yang hanya dijalankan sekali sesaat setelah mikrokontroler dijalankan atau di-*reset*. Merupakan bagian persiapan atau inisialisasi program.

2. *Void loop()*

Berisi kode program yang akan dijalankan terus-menerus. Merupakan untuk program utama

3. Instruksi percabangan *if* dan *if-else*

Instruksi *if* dan *if-else* akan menguji apakah kondisi tertentu dipenuhi atau tidak. Jika tidak dipenuhi, maka instruksi berikutnya akan dilompati, tetapi jika dipenuhi, maka instruksi berikutnya akan dijalankan.

4. Instruksi perulangan *for-loop*

Perulangan *for-loop* akan membuat perulangan pada bloknya dalam jumlah tertentu, yaitu sebanyak nilai counter-nya.

5. *Input Output* Digital

1. *pinMode()*

Ditempatkan di *void setup()*, digunakan untuk mengatur sebuah kaki I/O digital, untuk dijadikan *INPUT* atau *OUTPUT*, dengan format penulisan sebagai berikut :

```
pinMode(3,OUTPUT); // menjadikan D3 sebagai OUTPUT
```

2. *digitalRead()*

Digunakan untuk membaca sinyal digital yang masuk, digunakan instruksi *digitalRead()*, dengan format penulisan sebagai berikut :

```
int tombol=digitalRead(2); //membaca sinyal masuk di D2
```

3. *digitalWrite()*

Digunakan untuk mengeluarkan sinyal digital, dengan format penulisan sebagai berikut :

```
digitalWrite(3,HIGH); //mengeluarkan sinyal HIGH di D3
```

6. Komunikasi

1. Instruksi *Serial.begin()*

Digunakan untuk mengatur *baudrate* atau kecepatan komunikasi, umumnya nilainya adalah 9600

2. Instruksi *Serial.available()*

Digunakan untuk mendapatkan jumlah karakter atau *byte* yang telah diterima di serial port.

3. Instruksi *Serial.read()*

Digunakan untuk membaca data yang telah diterima di serial *port*

4. Instruksi *Serial.print()*

Digunakan untuk mencetak data ke serial *port*.

5. Instruksi *Serial.write()*

Digunakan untuk mengirimkan data dalam bentuk biner, satu *byte* data setiap pengiriman.

d. Fungsi Program Utama

```
#include <NewSoftSerial.h>
```

```
#define rxPin 2
```

```
#define txPin 3
```

```
NewSoftSerialrfid = NewSoftSerial( rxPin, txPin );
```

```
#define ledPin 13
```

```
#define unlockSeconds 2
```

Dalamblok proses ini merupakan inisialisasi awal program, penggunaan library dan pendefinisian pin yang digunakan/ konektor RFID dan mikrokontrolerl.

```
intnumberOfTags = sizeof(allowedTags)/sizeof(allowedTags[0]);
```

```
intincomingByte = 0;
```

Serial data yang masuk (tag value) disimpan di incomingbyte.

```
byte tagBytes[6];
```

```
char tagValue[10];
```

```
if((val = rfid.read()) == 2) {
```

```
bytesRead = 0;
```

```
while (bytesRead< 12) {
```

```
val = rfid.read();
```

```
if (bytesRead< 10)
```

```
{
```

```
tagValue[bytesRead] = val;
```

```
}
```

Proses program untuk membaca digit ID dari *tag card* yang masuk kemudian di cacah/ RFID scanning data.

```

else { // jikaselain database
Serial.println("Tag not authorized"); // cetak Tag not authorized
    }
Serial.println();
    }
rfid.flush(); // meresetisi data serial
bytesRead = 0; // byteReaddiisinol
    }
}

```

Kondisi program dimana tidak ada ID yang terbaca kesesuiannya dengan ID yang ada pada *database*.

```

void unlock() {
digitalWrite(ledPin, HIGH);
digitalWrite(futureOutput, HIGH);
    delay(unlockSeconds * 1000);
digitalWrite(futureOutput, LOW);
digitalWrite(ledPin, LOW);
}

```

Proses program saat ID cocok dengan *database*, mengaktifkan pin ledpin pada solenoid dan diberikan tunda selama 2 detik / solenoid aktif.

E. Pengujian Alat

Pengujian alat dilakukan untuk mendapatkan data penelitian. Dalam pengujian alat ini dilakukan dengan dua pengujian, yaitu :

1. Uji fungsional

Pengujian dilakukan dengan cara menguji setiap bagian alat berdasarkan karakteristik dan fungsi masing-masing. Pengujian dilakukan untuk mengetahui apakah setiap bagian dari perangkat telah bekerja sesuai dengan fungsi dan keinginan.

2. Uji unjuk kerja

Pengujian unjuk kerja alat dilakukan dengan cara melihat unjuk kerja alat. Hal-hal yang perlu diamati antara lain: rangkaian sistem minimum, rangkaian *driver* solenoid ke mikrokontroler, kerja sensor RFID, *tag card* dan solenoid. Dari pengujian ini akan diketahui kinerja dari alat yang dibuat.

F. Pengoperasian Alat

Pengoperasian alat ini dapat dilakukan dengan cara sebagai berikut:

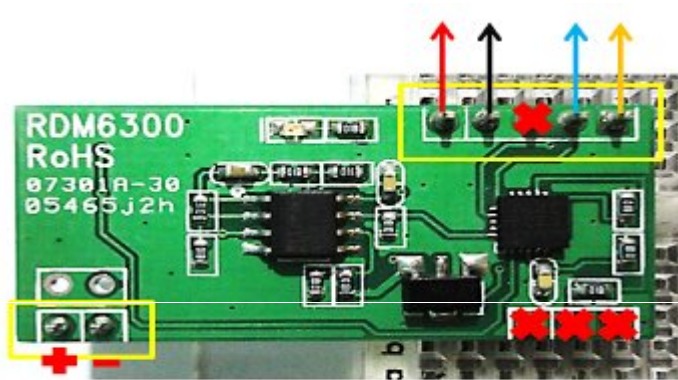
1. Pastikan alat terhubung dengan tegangan DC 12V yang sudah diturunkan menggunakan *IC* regulator 7805 menjadi 5V.
2. Hubungkan rangkaian sensor RFID, mikrokontroler ATmega 328, *driver* solenoid, dan solenoid.
3. Hadapkan *tag card* dengan sensor RFID untuk mengoperasikan pengaman pintu.

BAB IV

HASIL PENGUJIAN DAN PEMBAHASAN

A. Pengujian

Pengambilan data pada alat pengaman pintu rumah menggunakan RFID berbasis mikrokontroler ATmega328 ini dilakukan dengan pengamatan pada unjuk kerja desain penggunaan teknologi RFID, dilakukan pengukuran pada RFID yang digunakan sehingga dapat dihasilkan perbandingan antara teoritis dan secara praktiknya.



Gambar 27. Susunan pin RFID saat pengukuran

Warna kabel merah	=	Vcc
Warna kabel hitam	=	Ground
Warna kabel biru	=	Pin 3
Warna kabel oranye	=	Pin 2

1. Pengujian Sensor RFID

Pengambilan data pengukuran jarak dilakukan selama 5X pengambilan.

Tabel 2. Pengujian jarak sensor dengan *tag card*

Tipe Tag	Pengambilan Data Dan Jarak (cm)					Keterangan				
	D 1	D 2	D 3	D 4	D 5	D 1	D 2	D 3	D 4	D 5
Card	0	0	0	0	0	T	T	T	T	T
	0,5	0,5	0,5	0,5	0,5	T	T	T	T	T
	1	1	1	1	1	T	T	T	T	T
	1,5	1,5	1,5	1,5	1,5	T	T	T	T	T
	2	2	2	2	2	T	T	T	T	T
	2,5	2,5	2,5	2,5	2,5	T	T	T	T	T
	3	3	3	3	3	T	T	T	T	T
	3,5	3,5	3,5	3,5	3,5	T	T	T	T	T
	4	4	4	4	4	T	T	T	T	T
	4,5	4,5	4,5	4,5	4,5	D 3 S	D 3 S	D 3 S	D 3 S	D 3 S
	5	5	5	5	5	L	L	L	L	L
Gantungan kunci	0	0	0	0	0	T	T	T	T	T
	0,5	0,5	0,5	0,5	0,5	T	T	T	T	T
	1	1	1	1	1	T	T	T	T	T
	1,5	1,5	1,5	1,5	1,5	D 4 S	D 4 S	D 4 S	D 4 S	D 4 S
	2	2	2	2	2	L	L	L	L	L

Keterangan

T : Terbaca

L : Loss

D 3 S : *Delay 3 Second*

D 4 S : *Delay 4 Second*

D : Data



Gambar 28. *Tag* model gantungan kunci



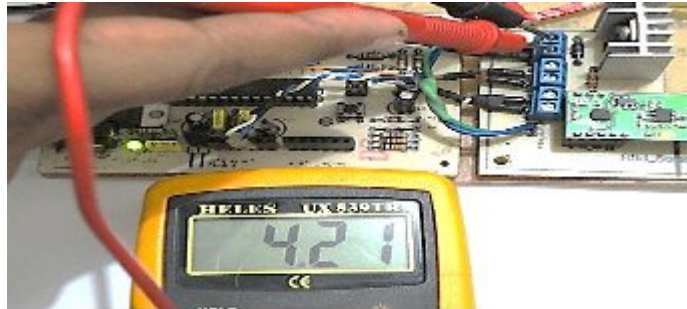
Gambar 29. *Tag* model *card* (kartu)

2. Pengujian Material Bahan Penghalang

Tabel 3. Material bahan penghalang yang dapat ditembus RFID *Reader*

Tipe Material	Kemampuan	
	Dapat ditembus	Tidak tertembus
Plastik	√	-
Kertas	√	-
Kain	√	-
Triplek	√	-
Box Karton	√	-
Aluminium	-	√
Besi	-	√
Seng	-	√

3. Pengujian Tegangan



Gambar 30. Pengukuran tegangan solenoid.

Pengukuran tegangan pada solenoid menunjukkan tegangan sebesar 4.21 volt. Sehingga diperoleh prosentase error sebesar :

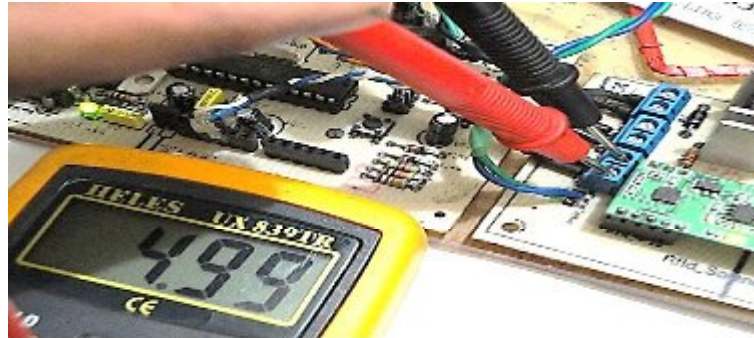
$$\frac{4,21 - 5}{5} \times 100\% = 15,8\%$$



Gambar 31. Pengukuran tegangan keluaran mikrokontroler pada pin digital 13 sebagai pemicu solenoid.

Pengukuran tegangan keluaran mikrokontroler sebagai pemicu solenoid menunjukkan tegangan 4.51 volt. Sehingga diperoleh prosentase error sebesar :

$$\frac{4,51 - 5}{5} \times 100\% = 9,8\%$$



Gambar 32. Pengukuran tegangan pada modul RFID.

Pengukuran tegangan pada modul RFID menunjukkan hasil tegangan sebesar 4,9 volt. Sehingga diperoleh prosentase error sebesar :

$$\frac{4,9 - 5}{5} \times 100\% = 2\%$$



Gambar 33. Pengukuran tegangan pada keluaran regulator 7805.

Pengukuran tegangan pada keluaran regulator 7805 menunjukkan tegangan sebesar 5 volt. Sehingga diperoleh prosentase error sebesar :

$$\frac{5 - 5}{5} \times 100\% = 0\%$$

Tabel 4. Hasil Pengukuran tegangan

Tegangan	Nilai	Prosentasi Eror
Solenoid	4,21 volt	$\frac{4,21 - 5}{5} \times 100\% = 15,8\%$
Pin 13	4,51 volt	$\frac{4,51 - 5}{5} \times 100\% = 9,8\%$
Modul RFID	4,99 volt	$\frac{4,9 - 5}{5} \times 100\% = 2\%$
Regulator 7805	5 volt	$\frac{5 - 5}{5} \times 100\% = 0\%$

4. Pengujian Solenoid

Table 5. Pengujian Solenoid

<i>Tag Card</i>	Satuan	Solenoid	
		Posisi mengunci	Posisi membuka
0	cm	√	-
1	cm	√	-
2	cm	√	-
3	cm	√	-
4	cm	√	-
5	cm	-	√

5. Pengujian Keseluruhan

Table 6. Pengujian keseluruhan

Tipe Tag	Jarak	Solenoid
<i>Card</i>	0 cm	Membuka pengunci
	0,5 cm	Membuka pengunci
	1 cm	Membuka pengunci
	1,5 cm	Membuka pengunci
	2 cm	Membuka pengunci
	2,5 cm	Membuka pengunci
	3 cm	Membuka pengunci
	3,5 cm	Membuka pengunci
	4 cm	Membuka pengunci
	4,5 cm	Membuka pengunci
	5 cm	<i>Loss</i>
Gantungan Kunci	0 cm	Membuka pengunci
	0,5 cm	Membuka pengunci
	1 cm	Membuka pengunci
	1,5 cm	Membuka pengunci
	2 cm	<i>Loss</i>

B. Pembahasan

1. Sensor RFID

Sensor RFID mampu membaca *tag card* dengan jara maksimal centi meter. Jarak maksimal dapat dideteksi pada jarak 4cm dengan *delay 3 second*, Jika menggunakan gantungan kunci jarak maksimal hanya 1.5cm dengan *delay 3 second*.

2. *Tag card*

Tag Card adalah alata yang digunaka nuntuk membuka pengunci dengan dihadapkan pada sensor RFID. Bahan-bahan yang dapat ditembus oleh RFID adalah bahan yang tidak terbuat dari logam dikarenakan semua benda yang berbahan material logam (besi,seng,aluminium,dll). Dikarenakan pembacaan RFID memanfaatkan model induksi dari lilitan yang ada pada antena RFID *reader*, jadi bila dihadapkan dengan material logam maka induksinya akan diserap, seperti pada magnet dan prinsip magnet.

3. Tegangan

Secara teoritis, keseluruhan rangkaian pengaman pintu yang berbasis mikrokontroler ATmega328 ini tersuplai menggunakan tegangan 12 volt dari adaptor yang diturunkan menggunakan *IC* regulator 7805 menjadi 5 volt. Dalam pengujian tegangan kali ini solenoid mendapatkan tegangan *output* 4,21 volt dengan prosentase error 15,8%. Tegangan Pin 13 mendapatkan 4,51 volt dengan prosentase error 9,8%. Tegangan Modul RFID mendapatkan 4,99 volt dengan prosentase error 2%, seharusnya semu rangkaian *input* maupun *output* mendapatkan 5 volt.

4. Solenoid

Solenoid yang berfungsi sebagai pengunci pintu dalam pengujian ini bisa berjalan dengan baik seperti yang diharapkan, dalam hal ini solenoid bisa mengunci dan membuka pengunci setelah *tag card* dihadapkan pada sensor FRID yang nantinya akan mengecek apakah *database* yang ada

pada nomer seri *tag card* sama pada yang ada di dalam mikrokontroler, apabil *databasenya* cocok maka solenoid akan bergerak membuka penguncinya. Solenoid beroperasi dengan menggunakan *timer*, setelah beberapa saat pintu terbuka maka solenoid akan kembali mengunci dengan otomatis.

C. Unjuk Kerja

Unjuk kerja alat ini merupakan penerapan dari diagram alir program utama, maka dari itu untuk mengoperasikan alat ini harus berpedoman dari diagram alir program utama.

Agar sistem dapat bekerja maka hal yang harus dilakukan adalah menghubungkan dengan tegangan sumber, kemudian sensor frid akan selalu memancarkan frekuensinya setelah mendapatkan inpiutan tegangan, apabila ada *tag card* yang berada disekitar sensor sesuai dalam ring kerjanya, maka ID yang ada pada *tag card* akan dikirim kemikrokontroler yang akan memprosea databesnya dari *tag card*. Bila cocok maka solenoid akan bergerak membuka penguncinya.

Pengaman pintu rumah menggunakan RFID bekerja pada Kemampuan pembacaan modul RFID terhadap *tag card* maksimal sejauh 4 cm dan terhadap *tag* berbentuk gantungan kunci sejauh 1 cm.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil pengujian dari alat pengaman rumah menggunakan RFID berbasis mikrokontroler ATmega328, maka dapat diambil kesimpulan sebagai berikut:

1. Perangkat keras sistem sebagai alat pengaman pintu rumah menggunakan RFID dapat diwujudkan dengan menggabungkan beberapa komponen dan rangkaian, diantaranya : rangkaian catu daya, sensor, rangkaian *driver* solenoid dan *output* (solenoid). Setiap rangkaian tersebut disatukan oleh mikrokontroler ATmega328 sebagai pusat kendali.
2. Perangkat lunak sistem sebagai alat pengaman pintu rumah menggunakan RFID dapat diwujudkan dengan menggunakan IDE Arduino 023. Pada pembuatan *listing program* menggunakan IDE Arduino 023 yang menggunakan *basic* bahasa C dengan penggabungan dari *Void setup()*, *Void loop()*, Instruksi percabangan *if* dan *if-else*, Instruksi perulangan *for-loop*, *Input Output Digital* ,Komunikasi.
3. Pengaman pintu rumah menggunakan RFID bekerja pada Kemampuan pembacaan modul RFID terhadap *tag card* maksimal sebesar 4 cm dan terhadap *tag* berbentuk gantungan kunci sebesar 1 cm. Dengan menggunakan sistem RFID sebagai sensor alat pengunci rumah, kita dapat meningkatkan kenyamanan dan keamanan untuk mengakses rumah tanpa

harus memegang bermacam-macam kunci yang mungkin sangat mengganggu dan dapat meningkatkan keamanan karena tidak semua pembobol pintu rumah mengerti cara membobol sistem ini.

4. Prosentase error pada pengukuran tegangan solenoid sebesar 15,8%, prosentase error pengukuran tegangan pin 13 sebesar 9,8%, prosentase error pengukuran tegangan pada modul RFID sebesar 2%, dan prosentase error pada regulator sebesar 0%.

B. Keterbatasan Alat

1. Sensor RFID tidak bias membaca *tag card* dengan jarak yang jauh.
2. Tidak adanya LCD pada alat pengaman rumah menggunakan RFID untuk penampil rekaman data yang masuk.
3. Tidak adanya buzzer pada Alat pengaman rumah menggunakan RFID untuk lebih meningkatkan keamanan.

C. Saran

Setelah melakukan pengujian terhadap kinerja dari pengaman rumah menggunakan kunci elektronik berbasis RFID, maka ada beberapa saran yang diberikan dari penulis untuk penggunaan dan kesempurnaan alat ini, yaitu:

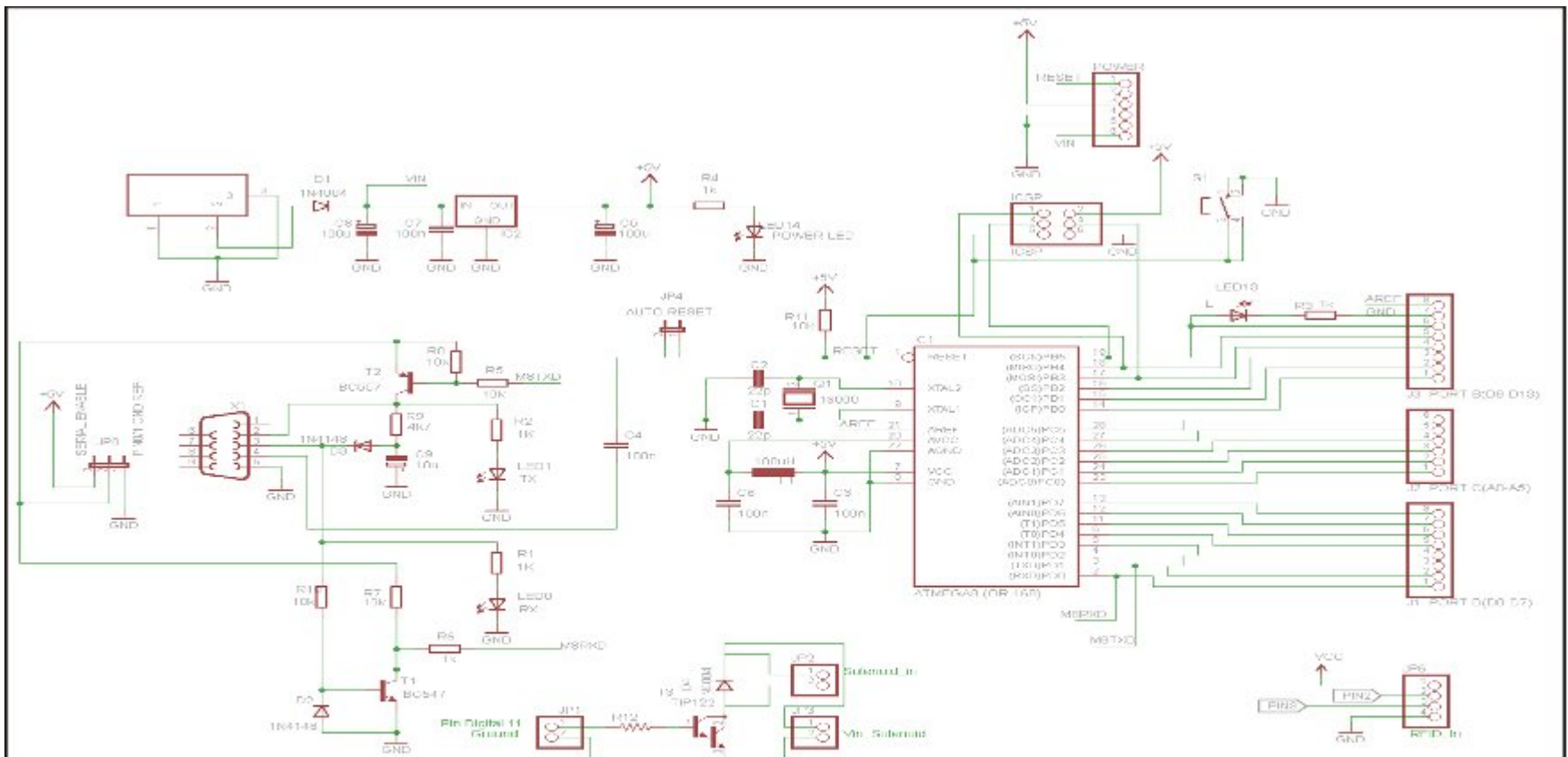
1. Menggunakan modul RFID yang lebih peka dalam kemampuan mengidentifikasi *tag card* untuk mengatasi *delay* yang terjadi.

2. Memberikan LCD supaya dapat merekam dan menampilkan *tag card* yang dibaca oleh sensor RFID.
3. Memberikan buzzer supaya bila ada *tag card* yang bukan ID yang ada didatabase maka alarem akan berunyi.

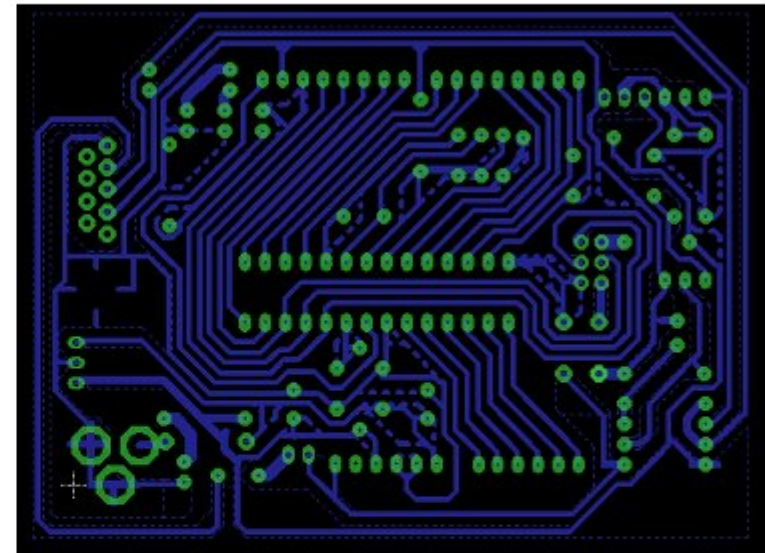
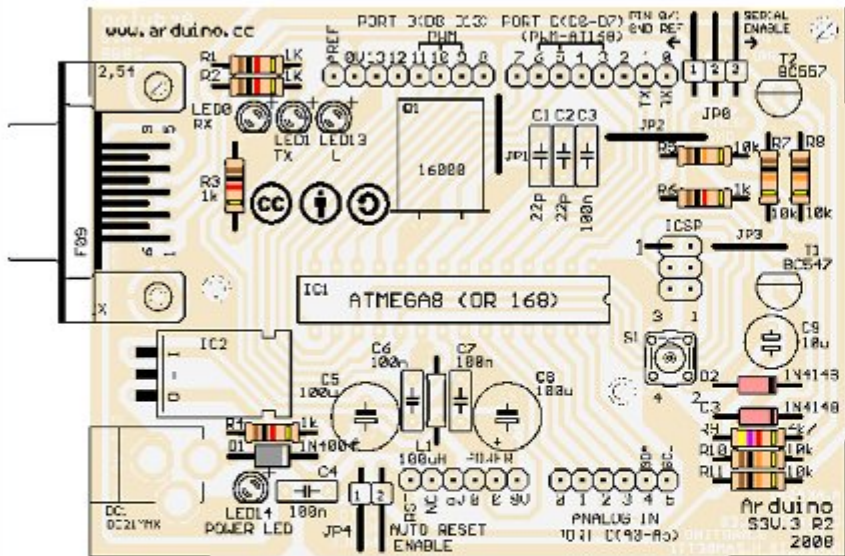
DAFTAR PUSTAKA

- Agfianto.Arduino. Diambil pada tanggal 5 April 2012, dari <http://agfi.staff.ugm.ac.id/blog/index.php/2010/08/arduino-duemilanove-dengan-atmega-328/>.
- Anonim. (2011). 3-Terminal *Positive Voltage Regulators*. Diambil pada tanggal 17 Maret 2012, dari : <http://www.national.com/mpf/LM/LM78M05.html>
- Johan Wijaya Kusuma 2009120003, Sigit Nugoho 200912000. Pengaman rumah menggunakan kunci elektronik berbasis rfid.
- McRoberts, Michael. 2010. *Beginning Arduino*. United States of America: Apress.
- Muhammad, Levy, 2009. Teknologi RFID pada perancangan sistem absensi dengan RFID menggunakan *custom RFID reader* menggunakan bahasa pemrograman java untuk *software* nya dan IC AT89S52.
- Mikrokontroler ATmega328. Diambil pada tanggal 10 Juli 2012, Dari www.atmel.com.
- Oxer, Jonathan and Blemings, Hugh. 2009. *Practical Arduino Cool Project for Open Source Hardware*. United States of America: Apress.
- Tim penyusun. 2011. Pendoman proyek Akhir. Fakultas Teknik UNY.
- Tom Igoe. 2007. *Making Things Talk*. United States of America: O'Reilly Media,inc.

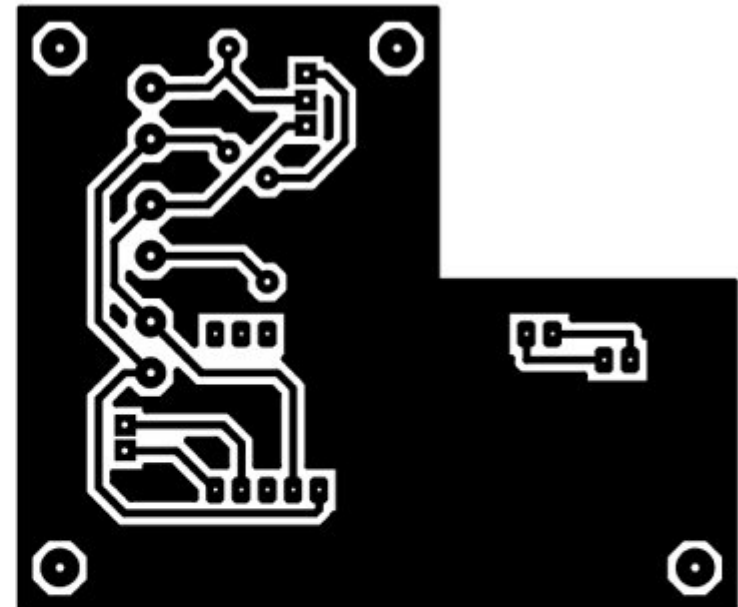
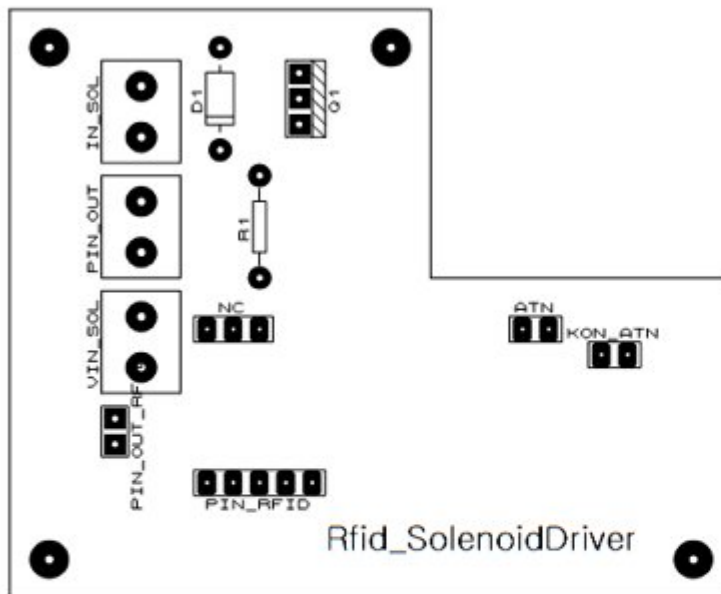
LAMPIRAN



LAYOUT DAN PCB RANGKAIAN			KETERANGAN	
FT UNY	SKALA : -	DIG: DIDIK	A4	No.1
	DIP. INDRI	DIST. INDRI	NIM. 09507131032	



LAYOUT DAN PCB RANGKAIAN			KETERANGAN	
			A4	No.2
FT UNY	SKALA : -	DIG: DIDIK		
		DIP. INDRI	DIST. INDRI	NIM. 09507131032



LAYOUT DAN PCB RANGKAIAN			KETERANGAN	
			A4	No.3
FT UNY	SKALA : -	DIG: DIDIK	NIM. 09507131032	
	DIP. INDRI	DIST. INDRI		

Bahan Dan Alat Yang Diperlukan Antara Lain :

- a) Perangkat komputer
 - b) PCB Fiber polos
 - c) FeCl_3
 - d) Bor
 - e) Solder
 - f) Aklirilk
 - g) Tenol
- 1) Proses perancangan rangkaian elektronik.
- a) Merancang gambar skematik rangkaian
 - b) Mendesain *Layout* PCB
 - c) Mencetak desain PCB
 - d) Melarutkan PCB menggunakan larutan FeCl_3
 - e) Mengebor PCB sesuai *layout* desain PCB
 - f) Memasang dan menyolder komponen ke papan PCB
 - g) Melakukan pengujian
- 2) Komponen yang di perlukan dalam perancangan rangkaian mikrokontroler ATmega328 adalah sebagai berikut :
- a) Capasitor Ceramic 22pF
 - b) *Capasitor Ceramic* 100nF
 - c) *Capasitor Elektrolic* 100 uF 16v
 - d) *Capasitor non polar elektrolic* 10uF 16v
 - e) Dioda 1N4004
 - f) Dioda 1N4148
 - g) *Jack* DC PCB
 - h) Regulator 7805
 - i) Induktor 100uH

- j) Led Hijau 3mm
- k) 16 MHz *Crystal*
- l) Resistor 1k 1/4 Watt
- m) Resistor 4k7 1/4 Watt
- n) Resistor 10k 1/4 Watt
- o) Transistor BC547 NPN
- p) Transistor BC557 PNP
- q) DB9 Pcb

Program Untuk Alat Pengaman Pintu Rumah Menggunakan RFID (*Radio Frequency Identification*) 125 khz Berbasis Mikrokontroler ATmega328

```
#include <NewSoftSerial.h>
#define rxPin 2
#define txPin 3
NewSoftSerialrfid = NewSoftSerial( rxPin, txPin );
#define ledPin 13
#define unlockSeconds 2
```

Dalamblok proses ini merupakan inisialisasi awal program, penggunaan library dan pendefinisian pin yang digunakan/ konektor RFID dan mikrokontrolerl.

```
char* allowedTags[] = {
    "04003B0E87",    // Tag value 1
    "04003B4E72",    // Tag value 2
};
```

Proses pengisian database ID card yang dapat diproses oleh mikrokontroler sebagai card yang valid di dalam program.

```
char* tagName[] = {
    "KARTU_1",    // Tag name 1
    "KARTU_2",    // Tag name 2
};
```

Proses pemberiannamadanpenyesuainnamauntuk database ID card sesuai dengan urutannya.

```
void setup() {
    pinMode(ledPin, OUTPUT);
    digitalWrite(ledPin, LOW);
    pinMode(futureOutput, OUTPUT);
    digitalWrite(futureOutput, LOW);
}
```

Proses inisialisasi program setup penggunaan pin dan mode logika keluaran pin yang akan dijalankan oleh program.

```
Serial.begin(9600);
rfid.begin(9600);
rfid.flush();
}
```

Proses membuka port dan kecepatan komunikasi serial dan menset nilai data serial untuk kembali reset setelah ada data yang masuk.

```
void loop() {
    byte i = 0; // nilaiawal byte i
    byte val = 0; // nilaiawal byte val
    byte checksum = 0; // nilaiawal byte checksum
    byte bytesRead = 0; // nilaiawalbyteRead
    byte tempByte = 0;
```

Proses program yang akan dijalankan secara berulang berupa nilai awal byte I, byte val, byte checksum, bytesRead, dan tempByte.

```

byte tagBytes[6];
char tagValue[10];
if((val = rfid.read()) == 2) {
bytesRead = 0;
while (bytesRead < 12) {
val = rfid.read();
if (bytesRead < 10)
{
tagValue[bytesRead] = val;
}
}
}

```

Proses program untuk membaca digit ID dari *tag card* yang masuk kemudian di cacah/ RFID scanning data.

```

if((val == 0x0D)|| (val == 0x0A)|| (val == 0x03)|| (val == 0x02)) {
break; }
if ((val >= '0') && (val <= '9')) {
val = val - '0';
}
else if ((val >= 'A') && (val <= 'F')) {
val = 10 + val - 'A';
}
}

```

Proses untuk membaca digit ID card hanya sampai 10 digit.

```

if (bytesRead & 1 == 1) {
tagBytes[bytesRead >> 1] = (val | (tempByte << 4));
if (bytesRead >> 1 != 5) {
checksum ^= tagBytes[bytesRead >> 1];
};
}
else {
tempByte = val;
};
bytesRead++;
}
if (bytesRead == 12) {
tagValue[10] = '\0';
Serial.print("Tag read: ");
for (i=0; i<5; i++) {
if (tagBytes[i] < 16) {
Serial.print("0"); //
}
Serial.print(tagBytes[i], HEX);
}
Serial.println();
Serial.print("Checksum: ");
Serial.print(tagBytes[5], HEX);
Serial.println(tagBytes[5] == checksum ? " -- passed." : " -- cocok.");
Serial.print("VALUE: ");
Serial.println(tagValue);
inttagId = findTag( tagValue );
}

```

Proses program untuk mengubah 10 digit ID menjadi 5 bit kemudian dirubah menjadi HEX dan dikeluarkan melalui port serial.

```

        if( tagId> 0 )
        {
            Serial.print("Authorized tag ID ");
            Serial.print(tagId);
            Serial.print(": unlocking for ");
            Serial.println(tagName[tagId - 1]);
            unlock();
        }

```

Kondisi syarat untuk mengeluarkan keterangan data yang masuk melalui *port* serial.

```

        else { // jikaselain database
            Serial.println("Tag not authorized"); // cetak Tag not authorized
        }
        Serial.println();
    }
    rfid.flush(); // meresetisi data serial
    bytesRead = 0; // byteReaddiisinol
}
}

```

Kondisi program dimana tidak ada ID yang terbaca kesesuiannya dengan ID yang ada pada *database*.

```

void unlock() {
    digitalWrite(ledPin, HIGH);
    digitalWrite(futureOutput, HIGH);
    delay(unlockSeconds * 1000);
    digitalWrite(futureOutput, LOW);
    digitalWrite(ledPin, LOW);
}

```

Proses program saat ID cocok dengan *database*, mengaktifkan pin ledpin pada solenoid dan diberikan tunda selama 2 detik / solenoid aktif.

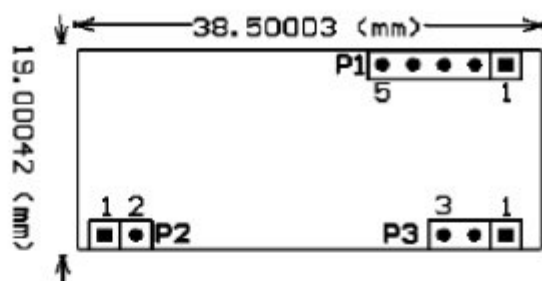
```

int findTag( char tagValue[10] ) {
    for (int thisCard = 0; thisCard<numberOfTags; thisCard++) {
        if(strcmp(tagValue, allowedTags[thisCard]) == 0)
        {
            return(thisCard + 1);
        }
    }
    return(0);
}

```

Proses program untuk mengecek kembali apakah ada data lagi yang masuk untuk ID pertama, jika tidak akan kembali pada proses awal.

Lampiran RDM630 RFID Modul

RDM630 Specification**1. Pin Definition (WEIGAND):****P1:**

PIN1	DATA0
PIN2	DATA1
PIN3	
PIN4	GND
PIN5	+5V(DC)

P2:

PIN1	ANT1
PIN2	ANT2

P3:

PIN1	LED
PIN2	+5V(DC)
PIN3	GND

2. Pin definition (TTL interface RS232 data format):**P1:**

PIN1	TX
PIN2	RX
PIN3	
PIN4	GND
PIN5	+5V(DC)

P2:

PIN1	ANT1
PIN2	ANT2

P3:

PIN1	LED
PIN2	+5V(DC)
PIN3	GND

Specification and Parameter:

Frequency	125KHz
Baud Rate	9600 (TTL Electricity Level RS232 format)
interface	Weigang26 Or TTL Electricity Level RS232 format
Power supply	DC 5V (± 5%)
Current	< 50Ma
Operating range	> 50mm (Depend on Card/Tag shape, manufacturer)
Expand I/O port	N/A
Indication light	N/A
Working temperature	-10℃ ~ +70℃
Storage temperature	-20℃ ~ +80℃
Max. humidity	Relative humidity 0 ~ 95%
Size	38.5mm×19mm×9mm

Wiegand Format 26bit output data format

Read only ID as 4 byte, form as below:-

D37	D36	D35	D34	D33	D32	D31	D30
D27	D26	D25	D24	D23	D22	D21	D20
D17	D16	D15	D14	D13	D12	D11	D10
D07	D06	D05	D04	D03	D02	D01	D00

Wiegand Format 26 bits format form by 26 bits data which including 24 bit user data and 2 bit parity bit. As far as module concerned, former 12bit of 24 bit data is even & the behind 12bits is odd.

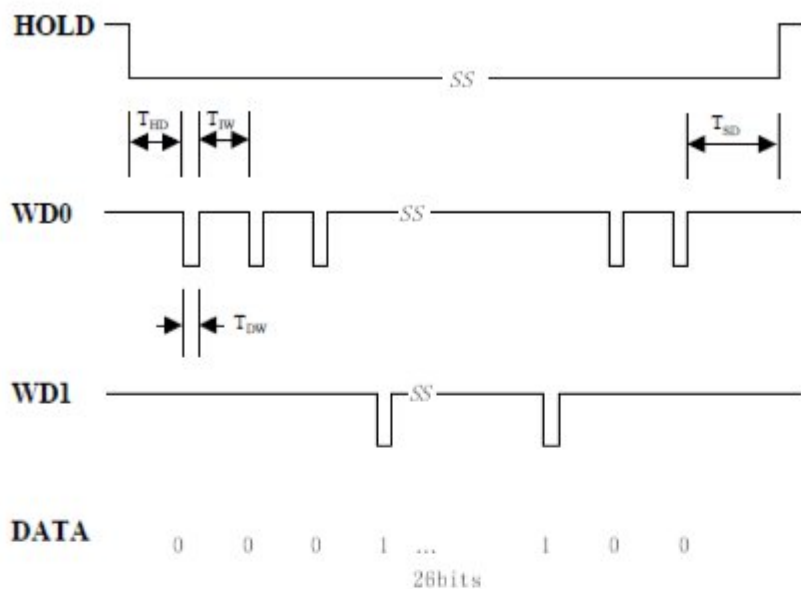
24 bit data correspond to 32 bit is read only 24bit which behind ID, I.E. D27-D20, D17-D10, D07-D00,

Output data format as chart below:

Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
	PE	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	PO
		E	E	E	E	E	E	E	E	E	E	E	O	O	O	O	O	O	O	O	O	O	O	O	O	
		D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	
		2	2	2	2	2	2	2	2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	
		7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	

Thereinto :

- ● PE is even bit, PO is odd bit;
- ● E is the data bit which was involved in even, O is the data bit which was involved in odd;
- ● DXX is the data bit which correspond to Mifare@ Standard & Light card read only ID;
- ● Wiegand Format 26bit output time sequence:



Symbol	Specification	Representative value
T_{HD}	Send data to active time extension	2ms
T_{SD}	Send data to finish time extension	2ms
T_{DW}	Data impulse width	80 μs
T_{IW}	Data impulse interval width	1ms

Reading Type:

Standard design -- card can't be read again within the range of read antenna after read once. User must move it out of reading antenna range first, swipe card again & can be read.

Special design -- card can be read continuously within the range of read antenna.

TTL Interface RS232 Data output format

1. 9600bps,N,8,1
2. CHECKSUM: card 10byte DATA entire do XOR operation

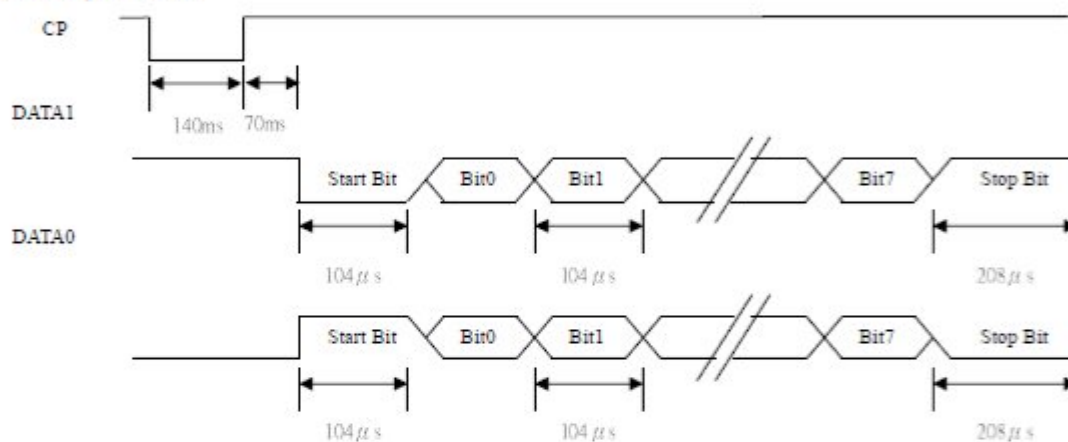
02	10ASCII Data Characters	Checksum	03
----	-------------------------	----------	----

Example: card number: 62E3086CED

Output data: 36H, 32H, 45H, 33H, 30H, 38H, 36H, 43H, 45H, 44H

CHECKSUM: (62H) XOR (E3H) XOR (08H) XOR (6CH) XOR (EDH) = 08H

2. Time sequence chart



Datasheet C7805



www.fairchildsemi.com

KA78XX/KA78XXA

3-Terminal 1A Positive Voltage Regulator

Features

- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

Description

The KA78XX/KA78XXA series of three-terminal positive regulators are available in the TO-220/DPAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.

TO-220

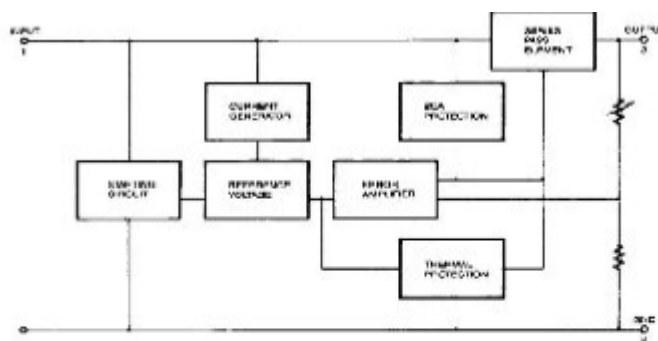


D-PAK



1. Input; 2. GND; 3. Output

Internal Block Diagram



Rev. 1.0.0

KA78XX/KA78XXA

Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$)	V_i	35	V
(for $V_O = 24V$)	V_i	40	V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}C/W$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	55	$^{\circ}C/W$
Operating Temperature Range (KA78XX(A/R))	T_{OPR}	$0 \sim +125$	$^{\circ}C$
Storage Temperature Range	T_{STG}	$-55 \sim +150$	$^{\circ}C$

Electrical Characteristics (KA7805/KA7805R)(Refer to test circuit, $0^{\circ}C < T_J < 125^{\circ}C$, $I_Q = 500mA$, $V_i = 10V$, $C_i = 0.33\mu F$, $C_o = 0.1\mu F$, unless otherwise specified.)

Parameter	Symbol	Conditions	KA7805			Unit
			Min.	Typ.	Max.	
Output Voltage	V_O	$T_J = +25^{\circ}C$	4.8	5.0	5.2	V
		$5.0mA < I_Q < 1.0A$, $P_Q < 15W$ $V_i = 7V$ to $20V$	4.75	5.0	5.25	
Line Regulation (Note1)	R_{glne}	$T_J = +25^{\circ}C$				mV
		$V_O = 7V$ to $25V$ $V_i = 8V$ to $12V$	-	4.0	100	
Load Regulation (Note1)	R_{glad}	$T_J = +25^{\circ}C$				mV
		$I_Q = 5.0mA$ to $1.5A$ $I_Q = 250mA$ to $750mA$	-	5	100	
Quiescent Current	I_Q	$T_J = +25^{\circ}C$	-	5.0	8.0	mA
Quiescent Current Change	ΔI_Q	$I_Q = 5mA$ to $1.0A$	-	0.03	0.5	mA
		$V_i = 7V$ to $25V$	-	0.3	1.3	
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_Q = 5mA$	-	0.8	-	mV/ $^{\circ}C$
Output Noise Voltage	V_{Ni}	$f = 10Hz$ to $100KHz$, $T_A = +25^{\circ}C$	-	42	-	$\mu V/V_O$
Ripple Rejection	RR	$f = 120Hz$ $V_O = 8V$ to $18V$	62	73	-	dB
Dropout Voltage	V_{Drop}	$I_Q = 1A$, $T_J = +25^{\circ}C$	-	2	-	V
Output Resistance	r_O	$f = 1KHz$		15		m Ω
Short Circuit Current	I_{SC}	$V_i = 35V$, $T_A = +25^{\circ}C$	-	230	-	mA
Peak Current	I_{PK}	$T_J = +25^{\circ}C$	-	2.2	-	A

Note:

1. Load and line regulation are specified at constant junction temperature. Changes in V_O due to heating effects must be taken into account separately. Pulse testing with low duty is used.

Lampiran TIP122

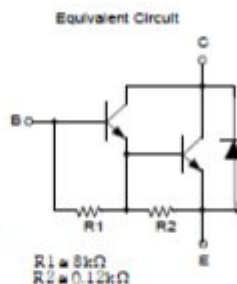


October 2008

TIP120/TIP121/TIP122

NPN Epitaxial Darlington Transistor

- Medium Power Linear Switching Applications
- Complementary to TIP125/126/127



Absolute Maximum Ratings* $T_a = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Ratings	Units
V_{CB0}	Collector-Base Voltage : TIP120	60	V
	: TIP121	80	V
	: TIP122	100	V
V_{CE0}	Collector-Emitter Voltage : TIP120	60	V
	: TIP121	80	V
	: TIP122	100	V
V_{EB0}	Emitter-Base Voltage	5	V
I_C	Collector Current (DC)	5	A
I_{CP}	Collector Current (Pulse)	8	A
I_B	Base Current (DC)	120	mA
P_C	Collector Dissipation ($T_a = 25^\circ\text{C}$)	2	W
	Collector Dissipation ($T_C = 25^\circ\text{C}$)	65	W
T_J	Junction Temperature	150	$^\circ\text{C}$
T_{STG}	Storage Temperature	- 65 ~ 150	$^\circ\text{C}$

* These ratings are limiting values above which the serviceability of any semiconductor device may be impaired.

Electrical Characteristics* $T_a = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Units
$V_{CEO(sus)}$	Collector-Emitter Sustaining Voltage : TIP120 : TIP121 : TIP122	$I_C = 100\text{mA}, I_B = 0$	60 80 100			V V V
I_{CEO}	Collector Cut-off Current : TIP120 : TIP121 : TIP122	$V_{CE} = 30\text{V}, I_B = 0$ $V_{CE} = 40\text{V}, I_B = 0$ $V_{CE} = 50\text{V}, I_B = 0$			0.5 0.5 0.5	mA mA mA
I_{CBO}	Collector Cut-off Current : TIP120 : TIP121 : TIP122	$V_{CB} = 60\text{V}, I_E = 0$ $V_{CB} = 80\text{V}, I_E = 0$ $V_{CB} = 100\text{V}, I_E = 0$			0.2 0.2 0.2	mA mA mA
I_{EBO}	Emitter Cut-off Current	$V_{BE} = 5\text{V}, I_C = 0$			2	mA
h_{FE}	* DC Current Gain	$V_{CE} = 3\text{V}, I_C = 0.5\text{A}$ $V_{CE} = 3\text{V}, I_C = 3\text{A}$	1000 1000			
$V_{CE(sat)}$	* Collector-Emitter Saturation Voltage	$I_C = 3\text{A}, I_B = 12\text{mA}$ $I_C = 5\text{A}, I_B = 20\text{mA}$			2.0 4.0	V V
$V_{BE(on)}$	* Base-Emitter On Voltage	$V_{CE} = 3\text{V}, I_C = 3\text{A}$			2.5	V
C_{ob}	Output Capacitance	$V_{CB} = 10\text{V}, I_E = 0, f = 0.1\text{MHz}$			200	pF

* Pulse Test: Pulse Width: 300 μ s, Duty Cycle: 2%

Typical characteristics

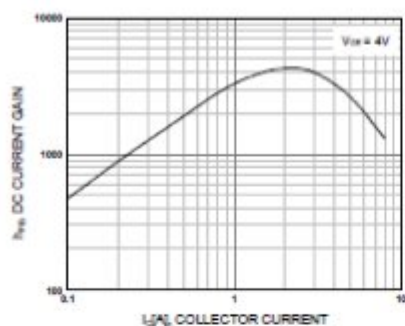


Figure 1. DC current Gain

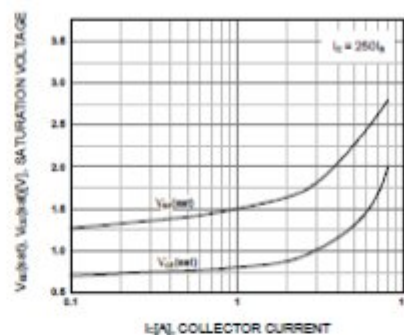
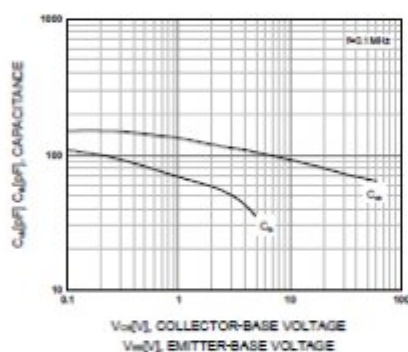
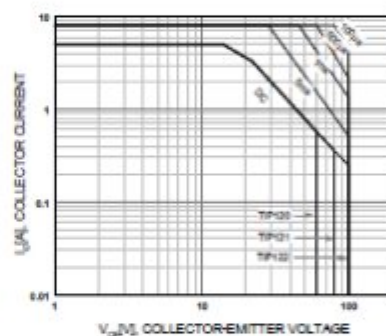
Figure 2. Base-Emitter Saturation Voltage
Collector-Emitter Saturation VoltageFigure 3. Output and Input Capacitance
vs. Reverse Voltage

Figure 4. Safe Operating Area

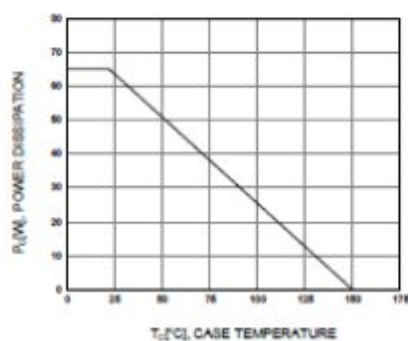
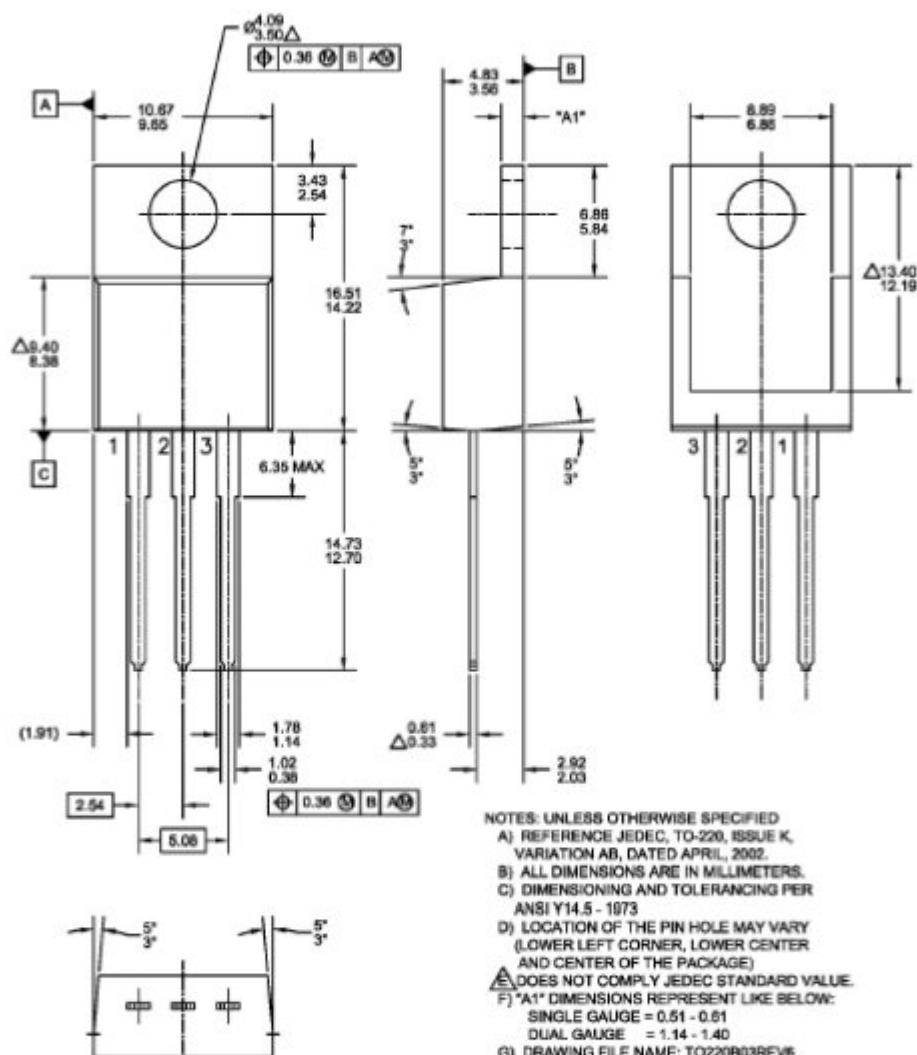


Figure 5. Power Derating

Mechanical Dimensions

TO220



NOTES: UNLESS OTHERWISE SPECIFIED

A) REFERENCE JEDEC, TO-220, ISSUE K, VARIATION AB, DATED APRIL, 2002.

B) ALL DIMENSIONS ARE IN MILLIMETERS.

C) DIMENSIONING AND TOLERANCING PER
ANSI Y14.5 - 1973

D) LOCATION OF THE PIN HOLE MAY VARY
(LOWER LEFT CORNER, LOWER CENTER)

AND CENTER OF THE PACKAGE)

DOES NOT COMPLY JEDEC STANDARD VALUE.

F) *A1* DIMENSIONS REPRESENT LIKE BELOW:

SINGLE GAUGE = 0.51 - 0.61

DUAL GAUGE = 1.14 - 1.40

G) DRAWING FILE NAME: TO220B03REV6



TRADEMARKS

The following are registered and unregistered trademarks and service marks Fairchild Semiconductor owns or is authorized to use and is not intended to be an exhaustive list of all such trademarks.

ACE [®]	Green FPS [™]	Power247 [®]	SuperSOT [™] -8
Build it Now [™]	Green FPS [™] e-Series [™]	POWEREDGE [®]	SyncFET [™]
CorePLUS [™]	GTO [™]	Power-SPM [™]	The Power Franchise [®]
CROSSVOLT [™]	i-Lo [™]	PowerTrench [®]	p[®]wer
CTL [™]	IntelliMAX [™]	Programmable Active Droop [™]	transistor
Current Transfer Logic [™]	ISOPANAR [™]	QFET [®]	TinyBoost [™]
EcoSPARK [®]	MegaBuck [™]	QS [™]	TinyBuck [™]
F[®]	MICROCOUPLER [™]	QT Optoelectronics [™]	TinyLogic [®]
Fairchild [®]	MicroFET [™]	Quiet Series [™]	TINYOPTO [™]
Fairchild Semiconductor [®]	MicroPak [™]	RapidConfigure [™]	TinyPower [™]
FACT Quiet Series [™]	MillerDrive [™]	SMART START [™]	TinyPWM [™]
FACT [®]	Motion-SPM [™]	SPM [®]	TinyWire [™]
FAST [®]	OPTOLOGIC [®]	STEALTH [™]	µSerDes [™]
FastvCore [™]	OPTOPLANAR [®]	SuperFET [™]	UHC [®]
FPS [™]	Q[®]	SuperSOT [™] -3	UniFET [™]
FRFET [®]	POP-SPM [™]	SuperSOT [™] -6	VCX [™]
Global Power Resource [™]	Power220 [®]		

DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION, OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS. THESE SPECIFICATIONS DO NOT EXPAND THE TERMS OF FAIRCHILD'S WORLDWIDE TERMS AND CONDITIONS, SPECIFICALLY THE WARRANTY THEREIN, WHICH COVERS THESE PRODUCTS.

LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR CORPORATION.

As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

PRODUCT STATUS DEFINITIONS

Definition of Terms

Datasheet Identification	Product Status	Definition
Advance Information	Formative or In Design	This datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	First Production	This datasheet contains preliminary data; supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice to improve design.
No Identification Needed	Full Production	This datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice to improve design.
Obsolete	Not In Production	This datasheet contains specifications on a product that has been discontinued by Fairchild semiconductor. The datasheet is printed for reference information only.

Rev. D31

Lampiran ATmega328

Features

- High Performance, Low Power AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 20 MIPS Throughput at 20 MHz
 - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 4/8/16/32K Bytes of In-System Self-Programmable Flash program memory
 - 256/512/1K Bytes EEPROM
 - 512/1K/1K/2K Bytes Internal SRAM
 - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/100 years at 25°C⁽¹⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Six PWM Channels
 - 8-channel 10-bit ADC in TQFP and QFN/MLF package
 - Temperature Measurement
 - 6-channel 10-bit ADC in PDIP Package
 - Temperature Measurement
 - Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Byte-oriented 2-wire Serial Interface (Philips I²C compatible)
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 23 Programmable I/O Lines
 - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating Voltage:
 - 1.8 - 5.5V
- Temperature Range:
 - -40°C to 85°C
- Speed Grade:
 - 0 - 4 MHz@1.8 - 5.5V, 0 - 10 MHz@2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
- Power Consumption at 1 MHz, 1.8V, 25°C
 - Active Mode: 0.2 mA
 - Power-down Mode: 0.1 µA
 - Power-save Mode: 0.75 µA (Including 32 kHz RTC)



8-bit AVR[®]
Microcontroller
with 4/8/16/32K
Bytes In-System
Programmable
Flash

ATmega48A
ATmega48PA
ATmega88A
ATmega88PA
ATmega168A
ATmega168PA
ATmega328
ATmega328P

Summary

Rev. 8271BS-AVR-04/10



ATmega48A/PA/88A/PA/168A/PA/328/P

1. Pin Configurations

Figure 1-1. Pinout ATmega48A/PA/88A/PA/168A/PA/328/P

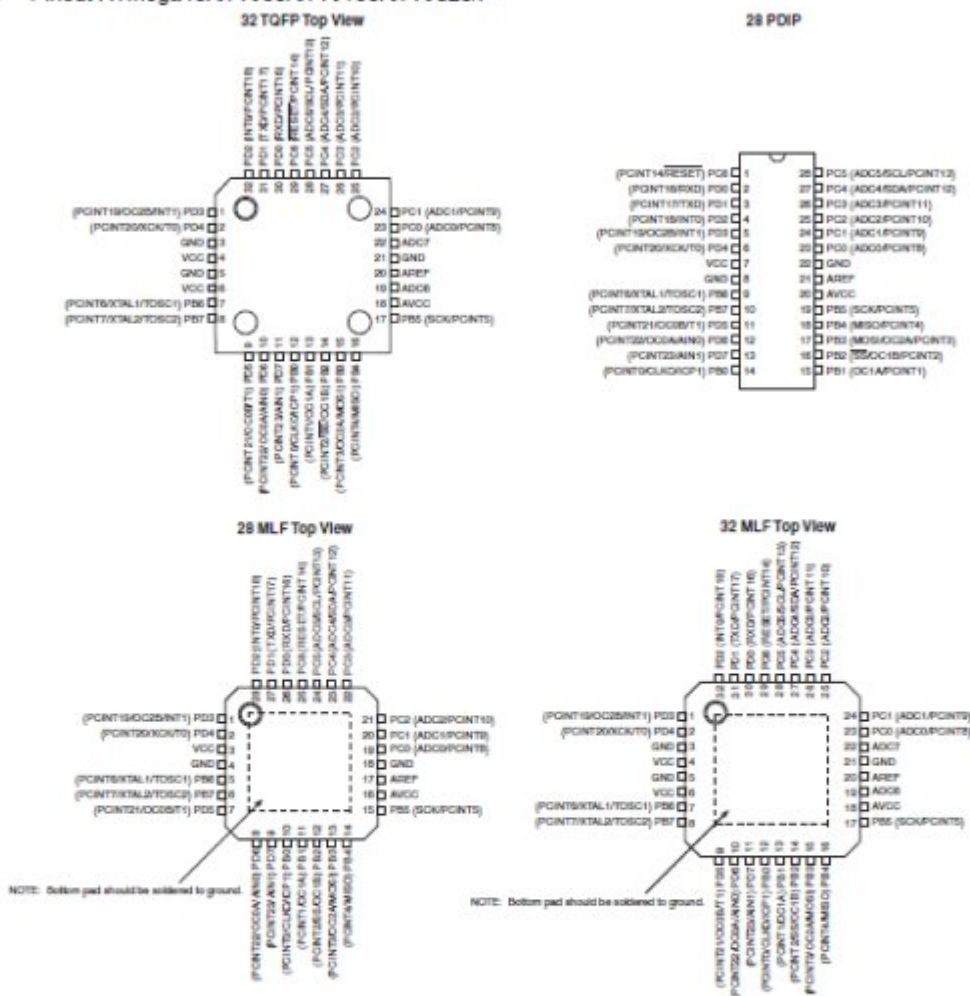


Table 1-1. 32UBGA - Pinout ATmega48A/48PA/88A/88PA/168A/168PA

	1	2	3	4	5	6
A	PD2	PD1	PC6	PC4	PC2	PC1
B	PD3	PD4	PD0	PC5	PC3	PC0
C	GND	GND			ADC7	GND
D	VDD	VDD			AREF	ADC6
E	PB6	PD6	PB0	PB2	AVDD	PB5
F	PB7	PD5	PD7	PB1	PB3	PB4

ATmega48A/PA/88A/PA/168A/PA/328/P

1.1 Pin Descriptions

1.1.1 VCC

Digital supply voltage.

1.1.2 GND

Ground.

1.1.3 Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7...6 is used as TOSC2...1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in ["Alternate Functions of Port B" on page 84](#) and ["System Clock and Clock Options" on page 27](#).

1.1.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5...0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

1.1.5 PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in [Table 29-12 on page 324](#). Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in ["Alternate Functions of Port C" on page 87](#).

1.1.6 Port D (PD7:0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

ATmega48A/PA/88A/PA/168A/PA/328/P

The various special features of Port D are elaborated in "[Alternate Functions of Port D](#)" on page 90.

1.1.7 AV_{CC}

AV_{CC} is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to V_{CC} , even if the ADC is not used. If the ADC is used, it should be connected to V_{CC} through a low-pass filter. Note that PC6...4 use digital supply voltage, V_{CC} .

1.1.8 AREF

AREF is the analog reference pin for the A/D Converter.

1.1.9 ADC7:6 (TQFP and QFN/MLF Package Only)

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

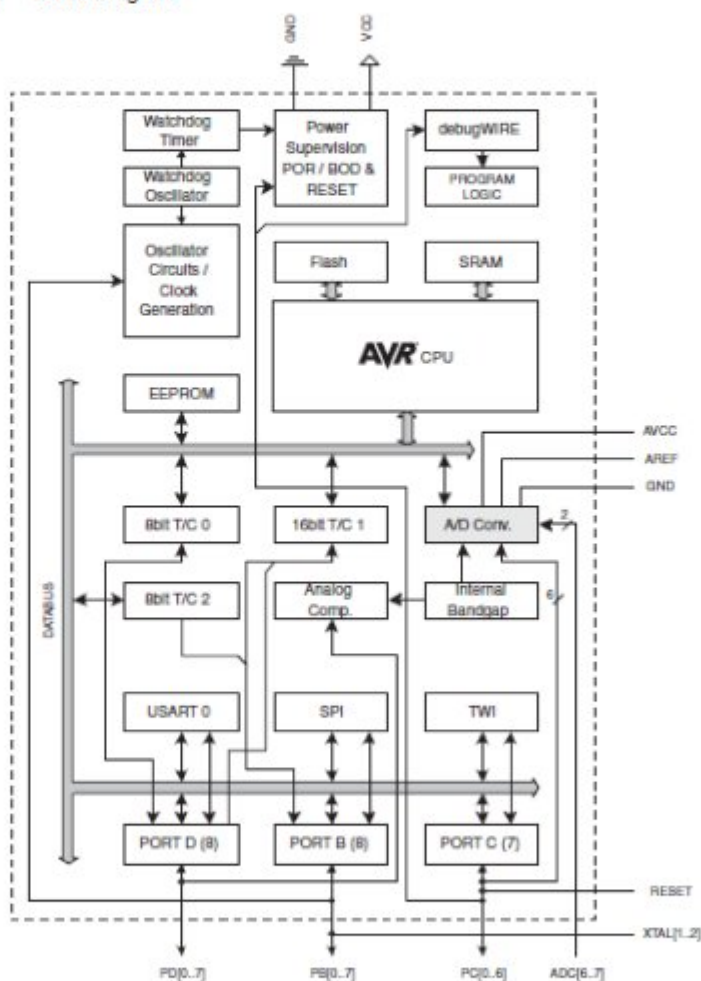
ATmega48A/PA/88A/PA/168A/PA/328/P

2. Overview

The ATmega48A/PA/88A/PA/168A/PA/328/P is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega48A/PA/88A/PA/168A/PA/328/P achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

2.1 Block Diagram

Figure 2-1. Block Diagram



The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting

ATmega48A/PA/88A/PA/168A/PA/328/P

architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The ATmega48A/PA/88A/PA/168A/PA/328/P provides the following features: 4K/8Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 256/512/512/1Kbytes EEPROM, 512/1K/1K/2Kbytes SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte-oriented 2-wire Serial Interface, an SPI serial port, a 6-channel 10-bit ADC (8 channels in TQFP and QFN/MLF packages), a programmable Watchdog Timer with internal Oscillator, and five software selectable power saving modes. The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, USART, 2-wire Serial Interface, SPI port, and interrupt system to continue functioning. The Power-down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next interrupt or hardware reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions. In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low power consumption.

Atmel® offers the QTouch® library for embedding capacitive touch buttons, sliders and wheels functionality into AVR® microcontrollers. The patented charge-transfer signal acquisition offers robust sensing and includes fully debounced reporting of touch keys and includes Adjacent Key Suppression® (AKS™) technology for unambiguous detection of key events. The easy-to-use QTouch Suite toolchain allows you to explore, develop and debug your own touch applications.

The device is manufactured using Atmel's high density non-volatile memory technology. The On-chip ISP Flash allows the program memory to be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip Boot program running on the AVR core. The Boot program can use any interface to download the application program in the Application Flash memory. Software in the Boot Flash section will continue to run while the Application Flash section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega48A/PA/88A/PA/168A/PA/328/P is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The ATmega48A/PA/88A/PA/168A/PA/328/P AVR is supported with a full suite of program and system development tools including: C Compilers, Macro Assemblers, Program Debugger/Simulators, In-Circuit Emulators, and Evaluation kits.

2.2 Comparison Between Processors

The ATmega48A/PA/88A/PA/168A/PA/328/P differ only in memory sizes, boot loader support, and interrupt vector sizes. Table 2-1 summarizes the different memory and interrupt vector sizes for the devices.

Table 2-1. Memory Size Summary

Device	Flash	EEPROM	RAM	Interrupt Vector Size
ATmega48A	4KBytes	256Bytes	512Bytes	1 instruction word/vector
ATmega48PA	4KBytes	256Bytes	512Bytes	1 instruction word/vector
ATmega88A	8KBytes	512Bytes	1KBytes	1 instruction word/vector

ATmega48A/PA/88A/PA/168A/PA/328/P

Table 2-1. Memory Size Summary (Continued)

Device	Flash	EEPROM	RAM	Interrupt Vector Size
ATmega88PA	8KBytes	512Bytes	1KBytes	1 instruction word/vector
ATmega168A	16KBytes	512Bytes	1KBytes	2 instruction words/vector
ATmega168PA	16KBytes	512Bytes	1KBytes	2 instruction words/vector
ATmega328	32KBytes	1KBytes	2KBytes	2 instruction words/vector
ATmega328P	32KBytes	1KBytes	2KBytes	2 instruction words/vector

ATmega48A/PA/88A/PA/168A/PA/328/P support a real Read-While-Write Self-Programming mechanism. There is a separate Boot Loader Section, and the SPM instruction can only execute from there. In ATmega 48A/48PA there is no Read-While-Write support and no separate Boot Loader Section. The SPM instruction can execute from the entire Flash.

ATmega48A/PA/88A/PA/168A/PA/328/P

8. AVR Memories

8.1 Overview

This section describes the different memories in the ATmega48A/PA/88A/PA/168A/PA/328/P. The AVR architecture has two main memory spaces, the Data Memory and the Program Memory space. In addition, the ATmega48A/PA/88A/PA/168A/PA/328/P features an EEPROM Memory for data storage. All three memory spaces are linear and regular.

8.2 In-System Reprogrammable Flash Program Memory

The ATmega48A/PA/88A/PA/168A/PA/328/P contains 4/8/16/32Kbytes On-chip In-System Reprogrammable Flash memory for program storage. Since all AVR instructions are 16 or 32 bits wide, the Flash is organized as 2/4/8/16K x 16. For software security, the Flash Program memory space is divided into two sections, Boot Loader Section and Application Program Section in ATmega88PA and ATmega168PA. See SELFPRGEN description in section ["SPMCSR – Store Program Memory Control and Status Register"](#) on page 295 for more details.

The Flash memory has an endurance of at least 10,000 write/erase cycles. The ATmega48A/PA/88A/PA/168A/PA/328/P Program Counter (PC) is 11/12/13/14 bits wide, thus addressing the 2/4/8/16K program memory locations. The operation of Boot Program section and associated Boot Lock bits for software protection are described in detail in ["Self-Programming the Flash, ATmega 48A/48PA"](#) on page 271 and ["Boot Loader Support – Read-While-Write Self-Programming"](#) on page 280. ["Memory Programming"](#) on page 297 contains a detailed description on Flash Programming in SPI- or Parallel Programming mode.

Constant tables can be allocated within the entire program memory address space (see the LPM – Load Program Memory instruction description).

Timing diagrams for instruction fetch and execution are presented in ["Instruction Execution Timing"](#) on page 14.

ATmega48A/PA/88A/PA/168A/PA/328/P

Figure 8-1. Program Memory Map ATmega 48A/48PA

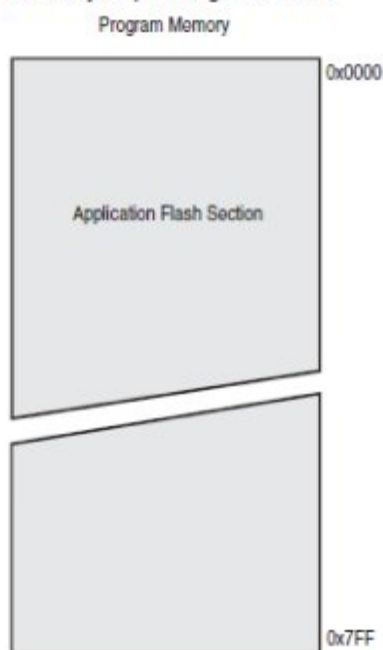
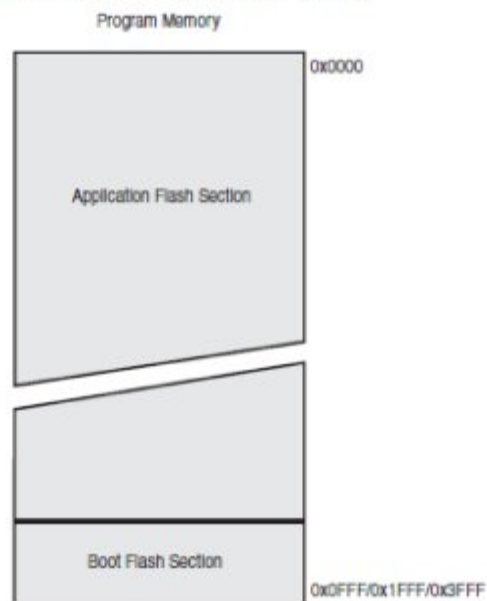


Figure 8-2. Program Memory Map ATmega88A, ATmega88PA, ATmega168A, ATmega168PA, ATmega328 and ATmega328P



ATmega48A/PA/88A/PA/168A/PA/328/P

8.3 SRAM Data Memory

Figure 8-3 shows how the ATmega48A/PA/88A/PA/168A/PA/328/P SRAM Memory is organized.

The ATmega48A/PA/88A/PA/168A/PA/328/P is a complex microcontroller with more peripheral units than can be supported within the 64 locations reserved in the Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 - 0xFF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

The lower 768/1280/1280/2303 data memory locations address both the Register File, the I/O memory, Extended I/O memory, and the internal data SRAM. The first 32 locations address the Register File, the next 64 location the standard I/O memory, then 160 locations of Extended I/O memory, and the next 512/1024/1024/2048 locations address the internal data SRAM.

The five different addressing modes for the data memory cover: Direct, Indirect with Displacement, Indirect, Indirect with Pre-decrement, and Indirect with Post-increment. In the Register File, registers R26 to R31 feature the indirect addressing pointer registers.

The direct addressing reaches the entire data space.

The Indirect with Displacement mode reaches 63 address locations from the base address given by the Y- or Z-register.

When using register indirect addressing modes with automatic pre-decrement and post-increment, the address registers X, Y, and Z are decremented or incremented.

The 32 general purpose working registers, 64 I/O Registers, 160 Extended I/O Registers, and the 512/1024/1024/2048 bytes of internal data SRAM in the ATmega48A/PA/88A/PA/168A/PA/328/P are all accessible through all these addressing modes. The Register File is described in "General Purpose Register File" on page 12.

Figure 8-3. Data Memory Map

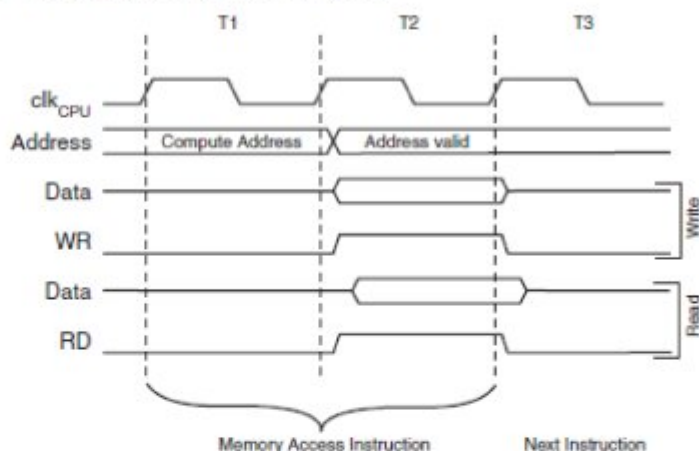
Data Memory	
32 Registers	0x0000 - 0x001F
64 I/O Registers	0x0020 - 0x005F
160 Ext I/O Reg.	0x0060 - 0x00FF
Internal SRAM (512/1024/1024/2048 x 8)	0x0100 0x02FF/0x04FF/0x4FF/0x08FF

ATmega48A/PA/88A/PA/168A/PA/328/P

8.3.1 Data Memory Access Times

This section describes the general access timing concepts for internal memory access. The internal data SRAM access is performed in two clk_{CPU} cycles as described in Figure 8-4.

Figure 8-4. On-chip Data SRAM Access Cycles



8.4 EEPROM Data Memory

The ATmega48A/PA/88A/PA/168A/PA/328/P contains 256/512/512/1Kbytes of data EEPROM memory. It is organized as a separate data space, in which single bytes can be read and written. The EEPROM has an endurance of at least 100,000 write/erase cycles. The access between the EEPROM and the CPU is described in the following, specifying the EEPROM Address Registers, the EEPROM Data Register, and the EEPROM Control Register.

"Memory Programming" on page 297 contains a detailed description on EEPROM Programming in SPI or Parallel Programming mode.

8.4.1 EEPROM Read/Write Access

The EEPROM Access Registers are accessible in the I/O space.

The write access time for the EEPROM is given in Table 8-2. A self-timing function, however, lets the user software detect when the next byte can be written. If the user code contains instructions that write the EEPROM, some precautions must be taken. In heavily filtered power supplies, V_{CC} is likely to rise or fall slowly on power-up/down. This causes the device for some period of time to run at a voltage lower than specified as minimum for the clock frequency used. See "Preventing EEPROM Corruption" on page 21 for details on how to avoid problems in these situations.

In order to prevent unintentional EEPROM writes, a specific write procedure must be followed. Refer to the description of the EEPROM Control Register for details on this.

When the EEPROM is read, the CPU is halted for four clock cycles before the next instruction is executed. When the EEPROM is written, the CPU is halted for two clock cycles before the next instruction is executed.

ATmega48A/PA/88A/PA/168A/PA/328/P

8.4.2 Preventing EEPROM Corruption

During periods of low V_{CC} , the EEPROM data can be corrupted because the supply voltage is too low for the CPU and the EEPROM to operate properly. These issues are the same as for board level systems using EEPROM, and the same design solutions should be applied.

An EEPROM data corruption can be caused by two situations when the voltage is too low. First, a regular write sequence to the EEPROM requires a minimum voltage to operate correctly. Secondly, the CPU itself can execute instructions incorrectly, if the supply voltage is too low.

EEPROM data corruption can easily be avoided by following this design recommendation:

Keep the AVR RESET active (low) during periods of insufficient power supply voltage. This can be done by enabling the internal Brown-out Detector (BOD). If the detection level of the internal BOD does not match the needed detection level, an external low V_{CC} reset Protection circuit can be used. If a reset occurs while a write operation is in progress, the write operation will be completed provided that the power supply voltage is sufficient.

8.5 I/O Memory

The I/O space definition of the ATmega48A/PA/88A/PA/168A/PA/328/P is shown in ["Register Summary" on page 533](#).

All ATmega48A/PA/88A/PA/168A/PA/328/P I/Os and peripherals are placed in the I/O space. All I/O locations may be accessed by the LD/LDS/LDD and ST/STS/STD instructions, transferring data between the 32 general purpose working registers and the I/O space. I/O Registers within the address range 0x00 - 0x1F are directly bit-accessible using the SBI and CBI instructions. In these registers, the value of single bits can be checked by using the SBIS and SBIC instructions. Refer to the instruction set section for more details. When using the I/O specific commands IN and OUT, the I/O addresses 0x00 - 0x3F must be used. When addressing I/O Registers as data space using LD and ST instructions, 0x20 must be added to these addresses. The ATmega48A/PA/88A/PA/168A/PA/328/P is a complex microcontroller with more peripheral units than can be supported within the 64 location reserved in Opcode for the IN and OUT instructions. For the Extended I/O space from 0x60 - 0xFF in SRAM, only the ST/STS/STD and LD/LDS/LDD instructions can be used.

For compatibility with future devices, reserved bits should be written to zero if accessed. Reserved I/O memory addresses should never be written.

Some of the Status Flags are cleared by writing a logical one to them. Note that, unlike most other AVRs, the CBI and SBI instructions will only operate on the specified bit, and can therefore be used on registers containing such Status Flags. The CBI and SBI instructions work with registers 0x00 to 0x1F only.

The I/O and peripherals control registers are explained in later sections.

ATmega48A/PA/88A/PA/168A/PA/328/P

8.5.1 General Purpose I/O Registers

The ATmega48A/PA/88A/PA/168A/PA/328/P contains three General Purpose I/O Registers. These registers can be used for storing any information, and they are particularly useful for storing global variables and Status Flags. General Purpose I/O Registers within the address range 0x00 - 0x1F are directly bit-accessible using the SBI, CBI, SBIS, and SBIC instructions.

8.6 Register Description

8.6.1 EEARH and EEARL – The EEPROM Address Register

Bit	15	14	13	12	11	10	9	8	
0x22 (0x42)	–	–	–	–	–	–	–	EEAR8	EEARH
0x21 (0x41)	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	R	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	X	
	X	X	X	X	X	X	X	X	

- Bits [15:9] – Reserved**

These bits are reserved bits in the ATmega48A/PA/88A/PA/168A/PA/328/P and will always read as zero.

- Bits 8:0 – EEAR[8:0]: EEPROM Address**

The EEPROM Address Registers – EEARH and EEARL specify the EEPROM address in the 256/512/512/1Kbytes EEPROM space. The EEPROM data bytes are addressed linearly between 0 and 255/511/511/1023. The initial value of EEAR is undefined. A proper value must be written before the EEPROM may be accessed.

EEAR8 is an unused bit in ATmega 48A/48PA and must always be written to zero.

8.6.2 EEDR – The EEPROM Data Register

Bit	7	6	5	4	3	2	1	0	
0x20 (0x40)	MSB							LSB	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- Bits 7:0 – EEDR[7:0]: EEPROM Data**

For the EEPROM write operation, the EEDR Register contains the data to be written to the EEPROM in the address given by the EEAR Register. For the EEPROM read operation, the EEDR contains the data read out from the EEPROM at the address given by EEAR.

8.6.3 EECR – The EEPROM Control Register

Bit	7	6	5	4	3	2	1	0	
0x1F (0x3F)	–	–	EEP1	EEP0	EERIE	EEMPE	EEPE	EERE	EECR
Read/Write	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	X	X	0	0	X	0	

- Bits 7:6 – Reserved**

These bits are reserved bits in the ATmega48A/PA/88A/PA/168A/PA/328/P and will always read as zero.

ATmega48A/PA/88A/PA/168A/PA/328/P

9.12 Register Description

9.12.1 OSCCAL – Oscillator Calibration Register

Bit	7	6	5	4	3	2	1	0	
(0x66)	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	Device Specific Calibration Value								

- Bits 7:0 – CAL[7:0]: Oscillator Calibration Value**

The Oscillator Calibration Register is used to trim the Calibrated Internal RC Oscillator to remove process variations from the oscillator frequency. A pre-programmed calibration value is automatically written to this register during chip reset, giving the Factory calibrated frequency as specified in [Table 29-10 on page 323](#). The application software can write this register to change the oscillator frequency. The oscillator can be calibrated to frequencies as specified in [Table 29-10 on page 323](#). Calibration outside that range is not guaranteed.

Note that this oscillator is used to time EEPROM and Flash write accesses, and these write times will be affected accordingly. If the EEPROM or Flash are written, do not calibrate to more than 8.8MHz. Otherwise, the EEPROM or Flash write may fail.

The CAL7 bit determines the range of operation for the oscillator. Setting this bit to 0 gives the lowest frequency range, setting this bit to 1 gives the highest frequency range. The two frequency ranges are overlapping, in other words a setting of OSCCAL = 0x7F gives a higher frequency than OSCCAL = 0x80.

The CAL6...0 bits are used to tune the frequency within the selected range. A setting of 0x00 gives the lowest frequency in that range, and a setting of 0x7F gives the highest frequency in the range.

9.12.2 CLKPR – Clock Prescale Register

Bit	7	6	5	4	3	2	1	0	
(0x61)	CLKPCE	–	–	–	CLKPS3	CLKPS2	CLKPS1	CLKPS0	CLKPR
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	See Bit Description				

- Bit 7 – CLKPCE: Clock Prescaler Change Enable**

The CLKPCE bit must be written to logic one to enable change of the CLKPS bits. The CLKPCE bit is only updated when the other bits in CLKPR are simultaneously written to zero. CLKPCE is cleared by hardware four cycles after it is written or when CLKPS bits are written. Rewriting the CLKPCE bit within this time-out period does neither extend the time-out period, nor clear the CLKPCE bit.

- Bits 3:0 – CLKPS[3:0]: Clock Prescaler Select Bits 3 - 0**

These bits define the division factor between the selected clock source and the internal system clock. These bits can be written run-time to vary the clock frequency to suit the application requirements. As the divider divides the master clock input to the MCU, the speed of all synchronous peripherals is reduced when a division factor is used. The division factors are given in [Table 9-17 on page 39](#).

ATmega48A/PA/88A/PA/168A/PA/328/P

10.11 Register Description

10.11.1 SMCR – Sleep Mode Control Register

The Sleep Mode Control Register contains control bits for power management.

Bit	7	6	5	4	3	2	1	0	
0x33 (0x53)	–	–	–	–	SM2	SM1	SM0	SE	SMCR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bits [7:4]: Reserved**

These bits are unused in the ATmega48A/PA/88A/PA/168A/PA/328/P, and will always be read as zero.

- **Bits 3:1 – SM[2:0]: Sleep Mode Select Bits 2, 1, and 0**

These bits select between the five available sleep modes as shown in [Table 10-2](#).

Table 10-2. Sleep Mode Select

SM2	SM1	SM0	Sleep Mode
0	0	0	Idle
0	0	1	ADC Noise Reduction
0	1	0	Power-down
0	1	1	Power-save
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Standby ⁽¹⁾
1	1	1	External Standby ⁽¹⁾

Note: 1. Standby mode is only recommended for use with external crystals or resonators.

- **Bit 0 – SE: Sleep Enable**

The SE bit must be written to logic one to make the MCU enter the sleep mode when the SLEEP instruction is executed. To avoid the MCU entering the sleep mode unless it is the programmer's purpose, it is recommended to write the Sleep Enable (SE) bit to one just before the execution of the SLEEP instruction and to clear it immediately after waking up.

ATmega48A/PA/88A/PA/168A/PA/328/P

10.11.2 MCUCR – MCU Control Register

Bit	7	6	5	4	3	2	1	0	
Ox25 (Ox55)	–	BODS ⁽¹⁾	BODSE ⁽¹⁾	PUD	–	–	IVSEL	IVCE	MCUCR
Read/Write	R	R/W	R/W	R/W	R	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 6 – BODS: BOD Sleep⁽¹⁾**

The BODS bit must be written to logic one in order to turn off BOD during sleep, see [Table 10-1 on page 40](#). Writing to the BODS bit is controlled by a timed sequence and an enable bit, BODSE in MCUCR. To disable BOD in relevant sleep modes, both BODS and BODSE must first be set to one. Then, to set the BODS bit, BODS must be set to one and BODSE must be set to zero within four clock cycles.

The BODS bit is active three clock cycles after it is set. A sleep instruction must be executed while BODS is active in order to turn off the BOD for the actual sleep mode. The BODS bit is automatically cleared after three clock cycles.

- **Bit 5 – BODSE: BOD Sleep Enable⁽¹⁾**

BODSE enables setting of BODS control bit, as explained in BODS bit description. BOD disable is controlled by a timed sequence.

Note: 1. BODS and BODSE only available for picoPower devices ATmega48PA/88PA/168PA/328P

10.11.3 PRR – Power Reduction Register

Bit	7	6	5	4	3	2	1	0	
(Ox64)	PRTWI	PRTIM2	PRTIM0	–	PRTIM1	PRSPI	PRUSART0	PRADC	PRR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

- **Bit 7 – PRTWI: Power Reduction TWI**

Writing a logic one to this bit shuts down the TWI by stopping the clock to the module. When waking up the TWI again, the TWI should be re initialized to ensure proper operation.

- **Bit 6 – PRTIM2: Power Reduction Timer/Counter2**

Writing a logic one to this bit shuts down the Timer/Counter2 module in synchronous mode (AS2 is 0). When the Timer/Counter2 is enabled, operation will continue like before the shutdown.

- **Bit 5 – PRTIM0: Power Reduction Timer/Counter0**

Writing a logic one to this bit shuts down the Timer/Counter0 module. When the Timer/Counter0 is enabled, operation will continue like before the shutdown.

- **Bit 4 – Reserved**

This bit is reserved in ATmega48A/PA/88A/PA/168A/PA/328/P and will always read as zero.

- **Bit 3 – PRTIM1: Power Reduction Timer/Counter1**

Writing a logic one to this bit shuts down the Timer/Counter1 module. When the Timer/Counter1 is enabled, operation will continue like before the shutdown.

ATmega48A/PA/88A/PA/168A/PA/328/P

- **Bit 2 – PRSPI: Power Reduction Serial Peripheral Interface**

If using debugWIRE On-chip Debug System, this bit should not be written to one.

Writing a logic one to this bit shuts down the Serial Peripheral Interface by stopping the clock to the module. When waking up the SPI again, the SPI should be re initialized to ensure proper operation.

- **Bit 1 – PRUSART0: Power Reduction USART0**

Writing a logic one to this bit shuts down the USART by stopping the clock to the module. When waking up the USART again, the USART should be re initialized to ensure proper operation.

- **Bit 0 – PRADC: Power Reduction ADC**

Writing a logic one to this bit shuts down the ADC. The ADC must be disabled before shut down. The analog comparator cannot use the ADC input MUX when the ADC is shut down.